NetIQ® CloudAccess and MobileAccess

Installation and Configuration Guide

March 2015

www.netiq.com/documentation

Legal Notice

THIS DOCUMENT AND THE SOFTWARE DESCRIBED IN THIS DOCUMENT ARE FURNISHED UNDER AND ARE SUBJECT TO THE TERMS OF A LICENSE AGREEMENT OR A NON-DISCLOSURE AGREEMENT. EXCEPT AS EXPRESSLY SET FORTH IN SUCH LICENSE AGREEMENT OR NON-DISCLOSURE AGREEMENT, NETIQ CORPORATION PROVIDES THIS DOCUMENT AND THE SOFTWARE DESCRIBED IN THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. SOME STATES DO NOT ALLOW DISCLAIMERS OF EXPRESS OR IMPLIED WARRANTIES IN CERTAIN TRANSACTIONS; THEREFORE, THIS STATEMENT MAY NOT APPLY TO YOU.

For purposes of clarity, any module, adapter or other similar material ("Module") is licensed under the terms and conditions of the End User License Agreement for the applicable version of the NetIQ product or software to which it relates or interoperates with, and by accessing, copying or using a Module you agree to be bound by such terms. If you do not agree to the terms of the End User License Agreement you are not authorized to use, access or copy a Module and you must destroy all copies of the Module and contact NetIQ for further instructions.

This document and the software described in this document may not be lent, sold, or given away without the prior written permission of NetIQ Corporation, except as otherwise permitted by law. Except as expressly set forth in such license agreement or non-disclosure agreement, no part of this document or the software described in this document may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, or otherwise, without the prior written consent of NetIQ Corporation. Some companies, names, and data in this document are used for illustration purposes and may not represent real companies, individuals, or data.

This document could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein. These changes may be incorporated in new editions of this document. NetIQ Corporation may make improvements in or changes to the software described in this document at any time.

U.S. Government Restricted Rights: If the software and documentation are being acquired by or on behalf of the U.S. Government or by a U.S. Government prime contractor or subcontractor (at any tier), in accordance with 48 C.F.R. 227.7202-4 (for Department of Defense (DOD) acquisitions) and 48 C.F.R. 2.101 and 12.212 (for non-DOD acquisitions), the government's rights in the software and documentation, including its rights to use, modify, reproduce, release, perform, display or disclose the software or documentation, will be subject in all respects to the commercial license rights and restrictions provided in the license agreement.

© 2015 NetIQ Corporation. All Rights Reserved.

For information about NetIQ trademarks, see https://www.netiq.com/company/legal/.

1 Overview of NetIQ CloudAccess 9

1.1 Inherent Problems Using SaaS Applications . . . 9
1.2 The Solution That CloudAccess Provides . . . 10
1.3 How CloudAccess Works . . . 10
1.4 How CloudAccess Fits into Your Network . . . 11
1.5 Understanding Product Licensing . . . 11

2 Installing the Appliance 13

2.1 Installation and Configuration Checklist . . . 13
2.2 Product Requirements . . . 14
2.3 Support for International Characters . . . 17
2.4 Identity Source Requirements . . . 17
2.4.1 Active Directory Requirements . . . 17
2.4.2 eDirectory Requirements . . . 18
2.4.3 JDBC Requirements . . . 19
2.4.4 Requirements for Other Identity Sources . . . 24
2.5 Appliance Installation Worksheet . . . 25
2.6 Deploying the Appliance . . . 26
2.7 Upgrading Your Environment . . . 26
2.8 Initializing the Appliance . . . 27

3 Configuring the Appliance 29

3.1 Accessing the Administration Console . . . 29
3.2 Getting Started . . . 29
3.3 Registering the Appliance . . . 30
3.4 Configuring Network Options . . . 30
3.4.1 Configuring the Forward Proxy . . . 31
3.4.2 Configuring the Second Network Interface . . . 31
3.4.3 Configuring the Routing Table . . . 32
3.4.4 A Sample Network Configuration . . . 32
3.5 Changing the Certificates on the Appliance . . . 33
3.6 Configuring Roles Management . . . 34
3.6.1 Defining the Role Types . . . 34
3.6.2 Assigning Roles to Users . . . 35
3.7 Configuring Clustering . . . 35
3.7.1 Advantages of Clustering . . . 35
3.7.2 Managing Nodes in the Cluster . . . 36
3.7.3 Configuring an L4 Switch for Clustering . . . 38
3.7.4 Configuring an L4 Switch for Email Proxy . . . 38
3.8 Configuring CloudAccess to Forward Events to a Syslog Server . . . 40

4 Configuring Identity Sources 41

4.1 Verifying the Identity Source User Attributes . . . 41
4.2 Configuring Additional Identity Sources . . . 41
4.2.1 Configuring Self-Service Registration and Password Management . . . 42
4.2.2 Configuring Additional Identity Sources . . . 47
4.2.3 Configuring SAML 2.0 Inbound Identity Sources . . . 47

5 Configuring Authentication Methods 49

5.1 Configuring Integrated Windows Authentication with Kerberos . . . 49
5.1.1 Configuring the Kerberos User in Active Directory . . . 49

5.1.2 Configuring the Appliance to Use Integrated Windows Authentication with Kerberos. 50
5.1.3 Configuring User Browsers 51
5.2 Configuring Google reCAPTCHA. 51
5.2.1 Requirements for reCAPTCHA 51
5.2.2 Configuring Intrusion Detection for Failed Logins 52
5.2.3 Configuring a Google reCAPTCHA Account 53
5.2.4 Configuring the reCAPTCHA Tool 53
5.3 Configuring the Time-Based One-Time Password (TOTP) Tool for Two-Factor Authentication Using Google Authenticator 54
5.3.1 Understanding One-Time Passwords 55
5.3.2 How to Use Google Authenticator for TOTP 56
5.3.3 Configuring the TOTP Tool 57
5.3.4 Registering a Mobile Device with the TOTP Tool for OTP Generation 58
5.3.5 Using Two-Factor Authentication at Login 60
5.3.6 Resetting a Device (Deregistering a Device). 60
5.4 Configuring the Advanced Authentication Tool for Two-Factor Authentication Using NetIQ Advanced Authentication Framework 61
5.4.1 Requirements for Advanced Authentication. 62
5.4.2 Configuring the Advanced Authentication Tool 62
5.5 Configuring FIDO for Two-Factor Authentication 63
5.5.1 Requirements for FIDO 63
5.5.2 Configuring the FIDO Tool 63
5.6 Configuring the Authentication Filter to Set Session-Based Identity Information for a User 64

6 Setting Up and Managing MobileAccess 67

6.1 Introduction to MobileAccess 67
6.2 Installing and Configuring the MobileAccess Appliance. 67
6.3 Configuring the MobileAccess Tool on the Appliance 68
6.4 Replacing the Default Certificate on the Appliance 68
6.4.1 Generating a Self-Signed Certificate 69
6.4.2 Installing a Self-Signed Certificate on the Mobile Device 69
6.5 Installing MobileAccess on a Mobile Device 70
6.6 Registering a Mobile Device with the Appliance 70
6.6.1 iOS Devices 70
6.6.2 Android Devices. 72
6.7 Understanding the MobileAccess PIN 73
6.8 Managing Mobile Devices 74
6.8.1 Unregistering Mobile Devices from the Administration Console 74
6.8.2 Unregistering a Mobile Device from the Device. 74
6.8.3 Deleting and Reinstalling the MobileAccess App on a Device. 75

7 Configuring Connectors 77

7.1 Overview of CloudAccess Connectors. 77
7.1.1 Understanding Single Sign-On Methods 78
7.1.2 Connectors for Federated Single Sign-On and Provisioning 82
7.1.3 Connectors for Federated Single Sign-On. 83
7.1.4 Connectors for Basic Single Sign-On 83
7.1.5 Connector for OAuth 2.0 Single Sign-On. 83
7.1.6 Connector for Simple Proxy Single Sign-On 84
7.1.7 Connector for Bookmarks 84
7.1.8 Application Catalog 84
7.1.9 Custom Connectors 84
7.1.10 License Information for Connectors. 85
7.2 Viewing Connectors for Applications 85
7.3 Providing Access to Applications for Users 85

7.4 How CloudAccess Provisions User Accounts . . . 86
7.4.1 Requirements for Provisioning . . . 86
7.4.2 Understanding Provisioning . . . 86
7.4.3 Samples of Account Creations . . . 87
7.4.4 CloudAccess Naming Convention for Newly Provisioned Accounts . . . 88
7.4.5 Matching Criteria for Merging Existing Accounts . . . 89
7.5 Configuring Appmarks for Connectors . . . 91
7.5.1 Understanding Appmark Options . . . 92
7.5.2 Mobile Device Workflow using Safari or Chrome . . . 93
7.5.3 Mobile Device Workflow with Internal Viewer . . . 94
7.5.4 Mobile Device Workflow from Bookmarks . . . 94
7.5.5 Configuring an Appmark for the Desktop Browser or Mobile Device . . . 94
7.5.6 Creating Multiple Appmarks for an Application . . . 95
7.5.7 Using Appmark Variables . . . 96
7.5.8 Policy Mapping for Non-Public Appmarks . . . 96
7.6 Connector for Google Apps for Business . . . 97
7.6.1 Connector Requirements . . . 97
7.6.2 Understanding Google Apps Provisioning . . . 97
7.6.3 Creating a Google Service Account . . . 98
7.6.4 Configuring the Connector for Google Apps . . . 99
7.6.5 Configuring Multiple Connectors for Google Apps . . . 100
7.6.6 Updating the Connector for Google Apps . . . 100
7.7 Connector for Microsoft Office 365 (SAML 2.0 or WS-Federation) . . . 101
7.7.1 How the Connector for Office 365 Works . . . 101
7.7.2 Connector Requirements . . . 103
7.7.3 Installing the Connector for Office 365 . . . 105
7.7.4 Validating the Connector for Office 365 . . . 106
7.7.5 Configuring Appmarks for Office 365 Applications . . . 106
7.7.6 Changing the Configuration of the Connector . . . 106
7.7.7 Changing the Name of an Office 365 Security Group . . . 106
7.7.8 Upgrading the Connector from SAML 2.0 to WS-Federation . . . 107
7.7.9 Uninstalling the Connector for Office 365 . . . 107
7.7.10 Installing Multiple Connectors for Office 365 . . . 107
7.8 Connector for Salesforce (SAML 2.0) . . . 108
7.8.1 Connector Requirements . . . 108
7.8.2 Configuring Salesforce to Trust CloudAccess . . . 109
7.8.3 Configuring the Connector for Salesforce . . . 109
7.8.4 Configuring Appmarks for Salesforce . . . 111
7.8.5 Configuring Multiple Connectors for Salesforce . . . 111
7.8.6 Using SSO to Salesforce on Mobile Devices . . . 112
7.8.7 Configuring Delegated Authentication in Salesforce . . . 115
7.8.8 Configuring the Salesforce Federation Identifier . . . 116
7.9 Connector for NetIQ Access Manager . . . 117
7.9.1 Requirements for the Connector for Access Manager . . . 117
7.9.2 Configuring the Connector . . . 117
7.9.3 Configuring Appmarks for Protected Resources in Access Manager . . . 118
7.10 Connector for Bookmarks . . . 118
7.10.1 Configuring the Connector for Bookmarks . . . 119
7.11 Connector for OAuth2 Resources . . . 120
7.11.1 Configuring the OAuth2 Client Application . . . 121
7.11.2 Configuring the Connector for OAuth2 Resources . . . 121
7.11.3 Supported OpenID Connect Schema . . . 122
7.12 Connector for Simple Proxy . . . 124
7.12.1 Requirements for Simple Proxy . . . 125
7.12.2 Viewing or Customizing the Attributes for Identity Injection . . . 125
7.12.3 Configuring the Connector for Simple Proxy . . . 128
7.13 Creating Custom Connectors . . . 130
7.13.1 Accessing the Access Connector Toolkit . . . 131
7.13.2 Toolkit Requirements . . . 132

7.13.3 Federation Requirements for the Application Service Provider . . . 132
7.13.4 Creating a SAML 2.0 Connector Template . . . 135
7.13.5 Creating a WS-Federation Connector Template . . . 137
7.13.6 Creating a SAML 2.0 Inbound (SAML2 In) Connector Template . . . 138
7.13.7 Creating a Basic SSO Connector Template . . . 142
7.13.8 Modifying a Connector . . . 144
7.13.9 Exporting a Connector Template . . . 144
7.13.10 Importing and Configuring Custom Connectors . . . 145
7.14 Downloading and Importing Single Sign-On Connectors . . . 148

8 Mapping Authorizations 151

8.1 Supported Roles and Authorizations . . . 151
8.2 Prerequisites . . . 151
8.3 Loading Authorizations . . . 152
8.4 Reloading Authorizations . . . 152
8.5 Mapping Authorizations . . . 152
8.6 Understanding Google Apps Mappings . . . 153
8.7 A Mapping Example . . . 154
8.8 Approving Requests . . . 154

9 Reporting 157

9.1 Using Google Analytics as an External Dashboard . . . 157
9.2 Integrating with Sentinel Log Manager . . . 157

10 Configuring the End User Experience 159

10.1 Configuring Email Clients . . . 159
10.2 Configuring End User Browsers for Kerberos Authentication . . . 159
10.3 Customizing Branding on User-Facing Pages . . . 160

11 Maintenance Tasks 163

11.1 Changing the Cluster Password . . . 163
11.2 Configuring Session Timeouts . . . 163
11.3 Changing the IP Address . . . 163
11.4 Changing the Public DNS Name or NTP Server Settings, or Uploading New Certificates . . . 164
11.5 Updating the Appliance . . . 164
11.6 Shutting Down or Rebooting a Node . . . 165
11.7 Recovering from a Disaster . . . 165

12 Troubleshooting CloudAccess 167

12.1 Troubleshooting the Appliance Initialization . . . 167
12.2 Displaying Health . . . 167
12.3 Using Troubleshooting Tools . . . 167
12.4 Troubleshooting Different States . . . 169
12.4.1 Master Node Health . . . 169
12.4.2 Front Panel of the Node . . . 169
12.4.3 Top of the Node . . . 170
12.4.4 Identity Source . . . 170
12.4.5 Applications . . . 171
12.4.6 Tools . . . 172
12.5 Troubleshooting Networking Issues . . . 173

12.6 Troubleshooting Provisioning Issues 175
12.6.1 Capturing Logs for Provisioning Issues 175
12.6.2 Understanding the Result of Actions Performed on Users and Groups. 175
12.7 Troubleshooting Policy Mapping Issues. 176
12.7.1 No Connectors Appear on the Policy Mapping Page 176
12.7.2 CloudAccess Does Not Reconcile Pending Approvals with Changes to Policy Mappings 177
12.7.3 Using Multiple Browsers or Browser Windows Can Result in Duplicate Mappings 177
12.8 Troubleshooting Mobile Device Issues. 177
12.8.1 Safari on Mobile Devices Cannot Access the Login Page After You Enable the MobileAccess Connector 177
12.8.2 Device Registration Issues 177
12.9 Troubleshooting CloudAccess Login Failures 178
12.10 Troubleshooting Authentications or Single Sign-On Issues 178
12.11 Troubleshooting Connector Issues. 178
12.12 Troubleshooting JDBC Identity Source Issues. 178

A Performing Advanced Branding 181

A.1 //Shared Include Files 181
A.2 //Standard Login Page 182
A.3 //Second Factor Authentication Login Pages 182
A.4 //Landing Page (Home Page after Login). 182
A.5 //Logout Page. 182

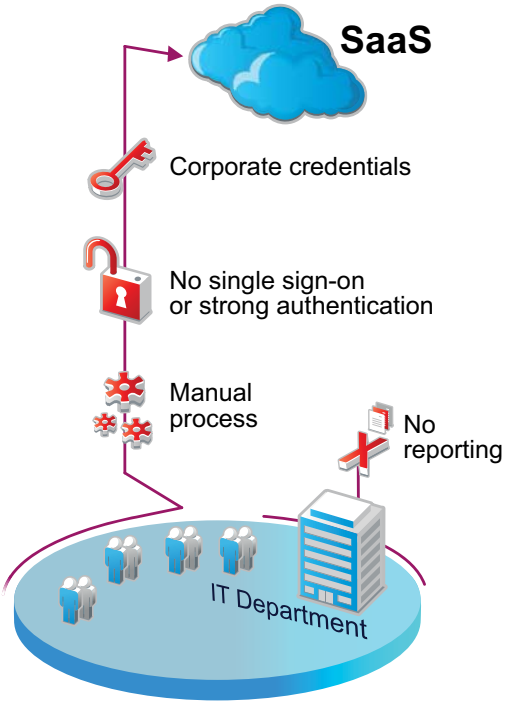
1 Overview of NetIQ CloudAccess

CloudAccess is a virtual appliance that enables you to provide secure access to Software-as-a-Service (SaaS) applications for your corporate users.

1.1 Inherent Problems Using SaaS Applications

Many corporate users want to use SaaS applications to increase business agility. If the corporation does not provide an easy way for users to obtain accounts for SaaS applications, several problems can occur. Figure 1-1 depicts some of these problems.

Figure 1-1 *Problems with Using SaaS Applications in the Corporation*



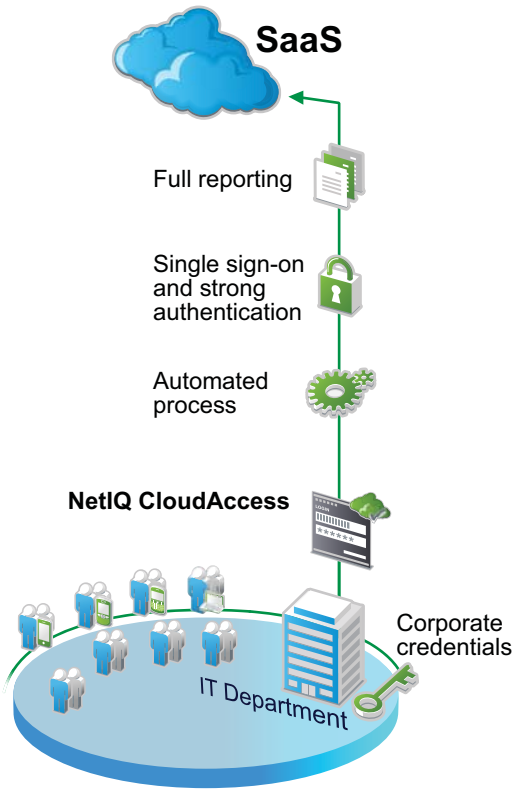
Common problems include the following:

- Users bypass the IT department and create their own accounts in the SaaS application.
- Users must wait for the IT department to create accounts in the SaaS application. It is a manual process, whether the IT department creates the account or the user creates the account.
- Users must remember separate passwords for each SaaS application, and often use their corporate credentials.
- Administrators receive no compliance reports of user activity in the SaaS application.

1.2 The Solution That CloudAccess Provides

CloudAccess provides a simple, secure solution to the problems presented with using SaaS applications.

Figure 1-2 *CloudAccess Solution*



CloudAccess provides the following benefits:

- An automated process to provision user accounts to the SaaS applications
- Secure single sign-on to the SaaS applications without corporate credentials leaving the security realm
- The ability for users to securely access SaaS applications inside or outside of the corporation
- Compliance reporting of users' activities in the SaaS applications

1.3 How CloudAccess Works

CloudAccess is a virtual appliance that provides a web service for users to securely access SaaS applications and other web applications. The appliance performs the following functions:

- **Provisioning:** CloudAccess allows you to map roles (groups) in an identity source, such as Active Directory or eDirectory, to account authorizations in the SaaS applications. CloudAccess leverages group management in the identity source to automatically create and manage the associated user accounts in the SaaS application.
- **Secure Single Sign-on:** CloudAccess authenticates users against identity sources and provides single sign-on to the SaaS applications and other web applications based on the users' entitlements. It also supports Integrated Windows Authentication (Kerberos) for background

authentication to CloudAccess. The corporate credentials never leave the firewall. Provisioned users automatically have access to the SaaS applications, if they are logged in to the identity source.

- **Reporting:** CloudAccess provides reports on the usage of the SaaS applications to help enforce corporate policies and prove compliance.
- **Enabling Mobile Devices:** CloudAccess enables mobile devices to securely access the SaaS applications.

NOTE: Whether you have a CloudAccess license or a MobileAccess-only license determines the application connectors you are entitled to use. For more information, see Section 1.5, "Understanding Product Licensing," on page 11.

1.4 How CloudAccess Fits into Your Network

CloudAccess resides behind the corporate firewall in your IT network. Administrators perform tasks in the console using a browser on a workstation inside or outside of the firewall.

You can cluster the CloudAccess appliance. By default, it is a single node cluster, but CloudAccess supports up to a five-node cluster. For more information about clustering, see Section 3.7, "Configuring Clustering," on page 35.

CloudAccess allows you to configure two network adapters for each node in the cluster. You can configure one adapter for the administrative network and a second adapter for the public network. For more information, see Section 3.4, "Configuring Network Options," on page 30.

The user accounts reside in the identity source. CloudAccess provisions those users to the SaaS application. Users can then access the SaaS application resources by logging in with their identity source accounts, whether they are inside or outside the firewall.

1.5 Understanding Product Licensing

If you purchased a full CloudAccess license, your license includes all of the MobileAccess features. Installing the CloudAccess appliance gives you all of the MobileAccess features automatically. For more information about enabling and configuring MobileAccess, see Chapter 6, "Setting Up and Managing MobileAccess," on page 67.

If you purchased MobileAccess without CloudAccess, your license entitles you to a 90-day trial of CloudAccess. At the end of that period, you are expected to purchase the appropriate license for CloudAccess or discontinue use of the CloudAccess features. Your MobileAccess license entitles you to use the following:

- All administrative features related to mobile device management
- All options on the Identity Sources palette
- All options on the Tools palette
- Three connectors on the Applications palette: the connector for NetIQ Access Manager, the Bookmarks connector, and the Simple Proxy connector

Under the MobileAccess license, you may not use any other embedded connectors (such as the connector for Salesforce or the connector for Google Apps) or import any other connectors (such as the connector for WebEx or custom connectors created with the Access Connector Toolkit).

MobileAccess-only customers can upgrade to a full CloudAccess license at any time. For more information about pricing, contact the NetIQ Sales Support team. For licensing purposes, you can upgrade to a full CloudAccess license by adding a single CloudAccess appliance to an existing MobileAccess cluster. However, if you want all nodes in the cluster to display the CloudAccess product name, you must manually replace each MobileAccess node with a CloudAccess node.

2 Installing the Appliance

The CloudAccess and MobileAccess products are both installed as a VMware virtual appliance using files that you download, extract, and deploy into your IT environment. Whether you have a MobileAccess-only license or a full CloudAccess license, you need to install the virtual appliance only once (in a non-clustered environment).

2.1 Installation and Configuration Checklist

Before you begin installing and configuring your appliance, review the following checklist to ensure that you perform steps in the appropriate order.

Table 2-1 *Installation and Configuration Checklist*

❑	Steps	For more information, see...
❑	1. Verify that your environment meets all prerequisites.	Section 2.2, “Product Requirements,” on page 14
❑	2. Verify that your identity source meets all requirements.	Section 2.4, “Identity Source Requirements,” on page 17
❑	3. Gather the information you need to install and configure the appliance.	Section 2.5, “Appliance Installation Worksheet,” on page 25
❑	4. Install the appliance.	Section 2.6, “Deploying the Appliance,” on page 26
❑	5. Initialize the appliance.	Section 2.8, “Initializing the Appliance,” on page 27
❑	6. Configure the appliance.	Chapter 3, “Configuring the Appliance,” on page 29
❑	7. Verify identity source user attributes.	Section 4.1, “Verifying the Identity Source User Attributes,” on page 41
❑	8. (Optional) Configure additional identity sources.	Chapter 4, “Configuring Identity Sources,” on page 41
❑	9. Configure authentication methods.	Chapter 5, “Configuring Authentication Methods,” on page 49
❑	10. Configure the MobileAccess tool on the appliance.	Section 6.3, “Configuring the MobileAccess Tool on the Appliance,” on page 68
❑	11. Replace the default certificate on the appliance.	Section 6.4, “Replacing the Default Certificate on the Appliance,” on page 68
❑	12. (Conditional) Install a self-signed or non-public certificate on the mobile device.	Section 6.4.2, “Installing a Self-Signed Certificate on the Mobile Device,” on page 69
❑	13. Determine which applications users need to be able to access from mobile devices.	

❑	Steps	For more information, see...
❑	14. Configure the appropriate connectors to enable user access to applications.	Chapter 7, “Configuring Connectors,” on page 77 *NetIQ® CloudAccess Connectors Guide*
❑	15. (Optional) Obtain or create custom icons in `.png` format to represent your appmarked applications.	
❑	16. Configure appmarks for the applications.	Section 7.5, “Configuring Appmarks for Connectors,” on page 91
❑	17. (Conditional) Map any non-public appmarks to the appropriate groups in the identity source.	Section 7.5.8, “Policy Mapping for Non-Public Appmarks,” on page 96
❑	18. (Conditional) If you have configured provisioning connectors, such as Google Apps, map authorizations for the SaaS applications.	Chapter 8, “Mapping Authorizations,” on page 151
❑	19. (Users) Install the MobileAccess app on their mobile devices.	Section 6.5, “Installing MobileAccess on a Mobile Device,” on page 70
❑	20. (Users) Register their mobile devices with the appliance.	Section 6.6, “Registering a Mobile Device with the Appliance,” on page 70

2.2 Product Requirements

Use the information in the following table to verify that your environment meets all requirements before deploying the appliance.

Table 2-2 *Product Requirements*

Components	Requirements

Supported Virtual Environments	The appliance requires one of the following virtual environments:
	◆ Hyper-V on Windows 2012 R2
	◆ One of the following versions of VMware:
	  ◆ ESXi 5.5
	  ◆ ESXi 5.1
	  ◆ ESXi 5.0 (U2 or later)

Components	Requirements
Virtual System Guest Requirements	Minimum hardware requirements for each appliance node in the cluster: ◆ 60 GB disk space ◆ 2 Cores ◆ 8 GB RAM The appliance can be a heavy consumer of CPU, disk I/O, and network bandwidth. Performance can be adversely affected by other virtual machines with similar operational requirements deployed on the same VMware host server. As a best practice, ensure that you group or separate virtual machines on hosts and data stores to avoid resource conflicts for CPU, disk I/O, and network bandwidth. You can do this manually as you deploy virtual machines, or use affinity and anti-affinity rules if they are available in your VMware environment.
Cluster	Supported cluster configuration: ◆ Up to a five-node cluster ◆ For optimal performance, each node should reside in the same IP subnet **NOTE:** The L4 switch must be configured with the publicly resolvable DNS of the cluster before you initialize the appliance.
Identity Source	Use one of the following identity sources for users: ◆ Microsoft Active Directory LDAP on Windows Server 2012 R2 or 2008 R2 ◆ NetIQ eDirectory LDAP 8.8 SP7 or 8.8 SP6 ◆ Microsoft SQL Server 2014 or 2008 ◆ Oracle Database 12c, 11.1, or 10.1 For more information, see Section 2.4, “Identity Source Requirements,” on page 17.
Browsers	**Administration:** Supported browsers for administration tasks: ◆ Firefox (latest version) on Windows 7 or 8.1 ◆ Google Chrome (latest version) on Windows 7 or 8.1 ◆ Internet Explorer (latest version) on Windows 7 or 8.1 ◆ Safari (latest version) on OS X Mavericks or later **Users:** Supported browsers for users: ◆ Firefox (latest version) on Windows 7 or 8.1 ◆ Google Chrome (latest version) on Windows 7 or 8.1 ◆ Internet Explorer (latest version) on Windows 7 or 8.1 ◆ Safari (latest version) on OS X Mavericks or later ◆ Safari or Chrome (latest version) on supported mobile devices **NOTE:** You must disable pop-up blockers to access the administration console. If you experience any issues with a supported browser, ensure that you have the latest version of the browser installed, or try another supported browser. Administering the appliance with Internet Explorer may be slower than with other supported browsers.

Components	Requirements
Mobile Devices	**Administration:** Not supported on mobile devices. **Users:** Supported iOS mobile devices for users: ◆ iPhone with iOS 7.0 or later ◆ iPad or iPad mini with iOS 7.0 or later Supported Android mobile devices for users: ◆ Android mid-density or high-density screen phone ◆ Android 10 inch or 7 inch tablet Supported Android versions: ◆ Ice Cream Sandwich 4.0 ◆ Jelly Bean 4.1 or later ◆ Kit Kat 4.4 and 5.0 **NOTE:** Only Kit Kat 4.4 and 5.0 support the login replay of previously stored credentials for the connectors for Basic Single Sign-On (SSO). On other Android platforms, users must log in manually to the destination websites.
Email Clients	For email proxy, CloudAccess supports IMAP, POP3, and SMTP across a variety of desktop and mobile email clients. For example, Windows Live Mail 2011 and the latest version of the Apple Mail Client on iPad or iPhone with iOS 7 or later. **NOTE:** The email ports in the CloudAccess cluster cannot be changed. It may be necessary to adjust the mail protocol or port configuration on the email clients to connect to the email proxy.
DNS	CloudAccess requires that all appliance nodes, administration workstations, end-user workstations, mobile devices, and identity sources be able to resolve the public DNS name of the appliance. The L4 switch must be configured with the publicly resolvable DNS of the cluster before you initialize the appliance.
SaaS Application Requirements	Each SaaS application has different requirements. For more information about the requirements for each SaaS application, see the *NetIQ® CloudAccess Connectors Guide*.

2.3 Support for International Characters

CloudAccess supports the use of international characters in values for identity attributes, except where the related identity source or application does not allow them. The international character can cause the authentication-related CloudAccess transaction to fail. For example, CloudAccess might be unable to provision a user's application account, to create a new account in an SSUS identity source, or to allow access to Simple Proxy accounts that use identity injection.

Table 2-3 identifies the identity attributes for external components that do not support using international characters.

Table 2-3 *Known Issues for International Characters in Identity Sources and Applications*

External Component	Does not support international characters for...
Identity Sources	
Active Directory	The email attribute value
Self Service User Store	The email address used as the user name for an account
Applications	
Any external OAUTH2 service	The user attribute
Google Apps for Business	The cn and sAMAccountName attribute values
Office 365	The cn and sAMAccountName attribute values
Salesforce	The email attribute value

2.4 Identity Source Requirements

Use the information in the following sections to verify that your identity source meets all requirements before you deploy the appliance. For CloudAccess to provision user accounts to the SaaS applications, each user account in the identity source must contain the attributes listed.

- Section 2.4.1, "Active Directory Requirements," on page 17
- Section 2.4.2, "eDirectory Requirements," on page 18
- Section 2.4.3, "JDBC Requirements," on page 19
- Section 2.4.4, "Requirements for Other Identity Sources," on page 24

2.4.1 Active Directory Requirements

Verify that your Active Directory environment meets the following requirements:

- ❒ Windows Server 2012 R2 or Windows Server 2008 R2.
- ❒ A unique identity for each user account, whether you have one or more domains or identity sources. The appliance uses the sAMAccountName as the unique identifier for the users.

To provision user accounts from Active Directory to the SaaS applications, all of the following attributes must be populated on the Active Directory users:

- First name

- Last name
- Full name (**Display name** is the field that populates this attribute.)
- sAMAccountName or Logon Name (Pre-Windows 2000)
- User Principal Name (UPN)
- Email address

Obtain the following required items:

- The password and the fully distinguished LDAP-formatted name of a user in Active Directory who has read access to the user objects. The appliance will use this user account to make LDAP binds to Active Directory.
- The name and password of a user in Active Directory who becomes the administrator of the appliance. The user must reside in the user search context specified during the appliance initialization procedure.
- The IP address of one or more Active Directory servers that contain the users.
- The context of the users in Active Directory.

2.4.2 eDirectory Requirements

Verify that your eDirectory environment meets the following requirements:

- ❒ eDirectory 8.8 SP7 or eDirectory 8.8 SP6.
- ❒ A unique identity for each user account, whether you have one or more eDirectory trees or identity sources.

To provision user accounts from eDirectory to the SaaS applications, all of the following attributes must be populated on the eDirectory users:

- CN (**Username** is the field that populates this attribute.)
- Given Name (**First name** is the field that populates this attribute.)
- Internet EMail Address
- Surname (**Last name** is the field that populates this attribute.)

Obtain the following required items:

- The password and fully distinguished LDAP-formatted name of a user in eDirectory who has the following rights. The appliance will use this user account to make LDAP binds to eDirectory:
  - **Property Rights**
    - **CN:** compare, read, inherit
    - **Description:** compare, read, inherit
    - **Given Name:** compare, read, inherit
    - **GUID:** compare, read, inherit
    - **Internet EMail Address:** compare, read, inherit
    - **Login Disabled:** compare, read, inherit
    - **Member:** compare, read, inherit
    - **Group Membership:** compare, read, inherit
    - **Surname:** compare, read, inherit
  - **Entry Rights:** browse, inherit

- The name and password of a user in eDirectory who becomes the administrator of the appliance. The user must reside in the subtree of the search context for the identity source specified during the initialization of the appliance.
- The IP address of one or more eDirectory servers that contain a replica of the partition holding the user objects and that run NLDAP.
- The context of the users in eDirectory.

2.4.3 JDBC Requirements

In order to use the JDBC database as an identity source, you must know and understand JDBC databases. The information provided in this section is intended for database administrators.

- "Meeting the Requirements" on page 19
- "Obtaining the Script Files" on page 19
- "Populating the Required Columns" on page 20
- "Dataflow Information" on page 21

Meeting the Requirements

Verify that you meet the following requirements or obtain the following information before using a JDBC database as an identity source:

- The supported type of JDBC database. For more information, see "Identity Source" on page 15.
- The IP address of the JDBC database.
- The port for communication. The default port is 1433 for Microsoft SQL Server or 1521 for Oracle Database.
- The database name or sid. (`idm` for Microsoft SQL Server defined as the `sid` in Oracle Database).
- The script files must be installed before you can configure a JDBC database as an identity source. For more information, see "Obtaining the Script Files" on page 19.
- The password for the user name in the sample scripts you install. For more information, see "Obtaining the Script Files" on page 19.

Obtaining the Script Files

In order to use JDBC as an identity source, you must install script files on your JDBC database so CloudAccess knows what tables to read to access the users and groups information. You download the script files when you configure the JDBC identity source.

You can download the files during the initialization process, if you select JDBC as your identity source, or you can download the scripts when you configure JDBC as an identity source after the appliance is initialized. You download a single zipped file that contains multiple scripts.

The different scripts are:

- **indirect_install:** Installs the schema, which includes the indirect tablespace and `proc_authuser ()` stored procedure, as well as the automatic triggers for the `indirect.user` and `indirect.grp` tables.
- **copy_from:** Copies user account information from the database default user store into the `indirect.usr` table for processing by the connector for JDBC.

- **uninstall:** Removes the schema and deletes or drops the connector user accounts in the underlying database.

Populating the Required Columns

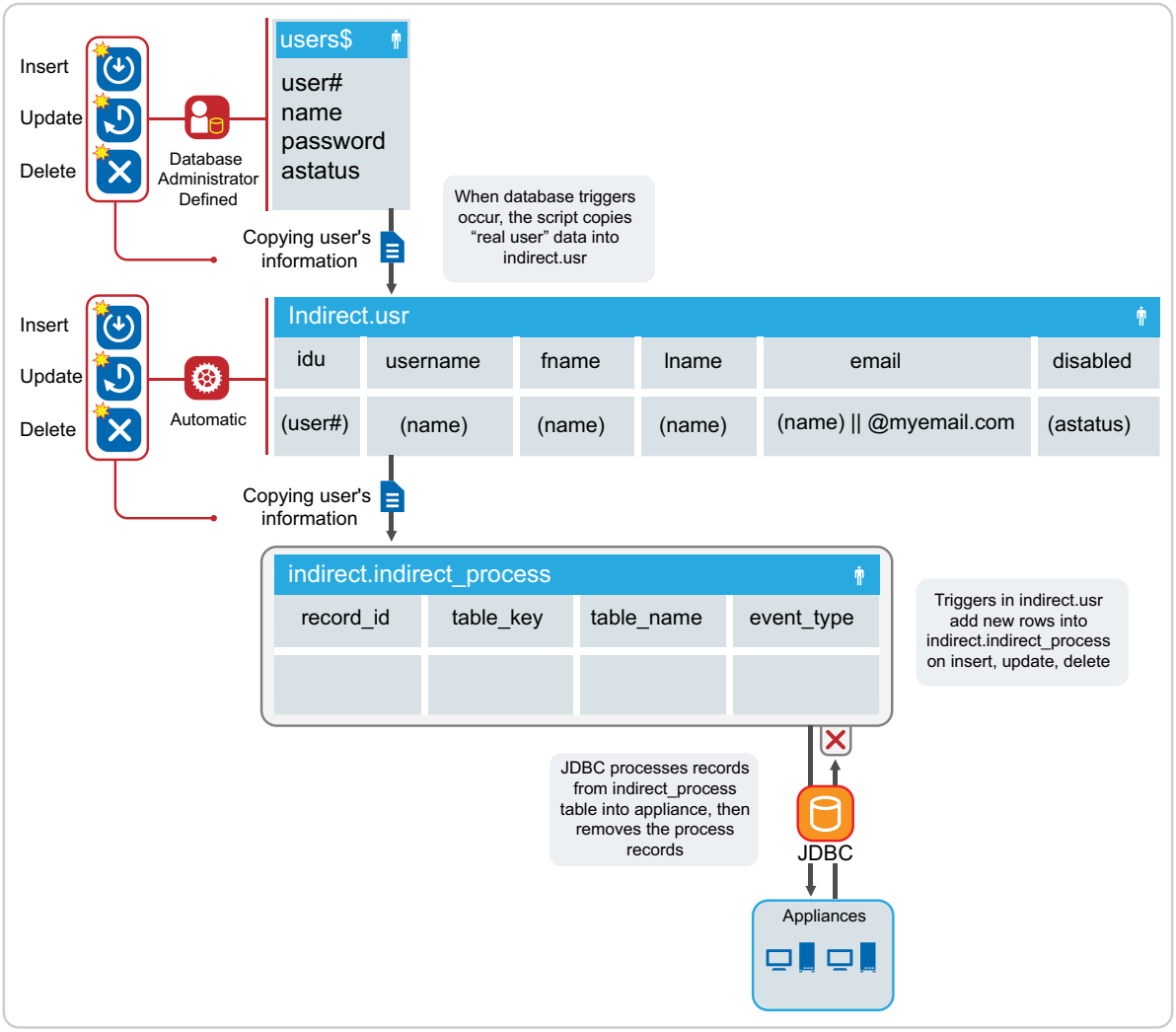
To provision users from the JDBC databases to the SaaS applications, you must have the following columns populated for each user account in the JDBC database:

- `indirect.usr.idu`
- `indirect.usr.username`
- `indirect.usr.fname` (Mandatory only for Google Apps accounts)
- `indirect.usr.lname`
- `indirect.usr.email` (Mandatory only for Salesforce accounts)

Dataflow Information

The connector for JDBC uses indirect tables to gather the needed information, ensuring that the appliance does not work directly with the information in the database. The following graphic depicts how the connector for JDBC obtains the information from the JDBC database. This process is the same no matter what type of database the appliance connects to.

Figure 2-1 *Dataflow of User Information*



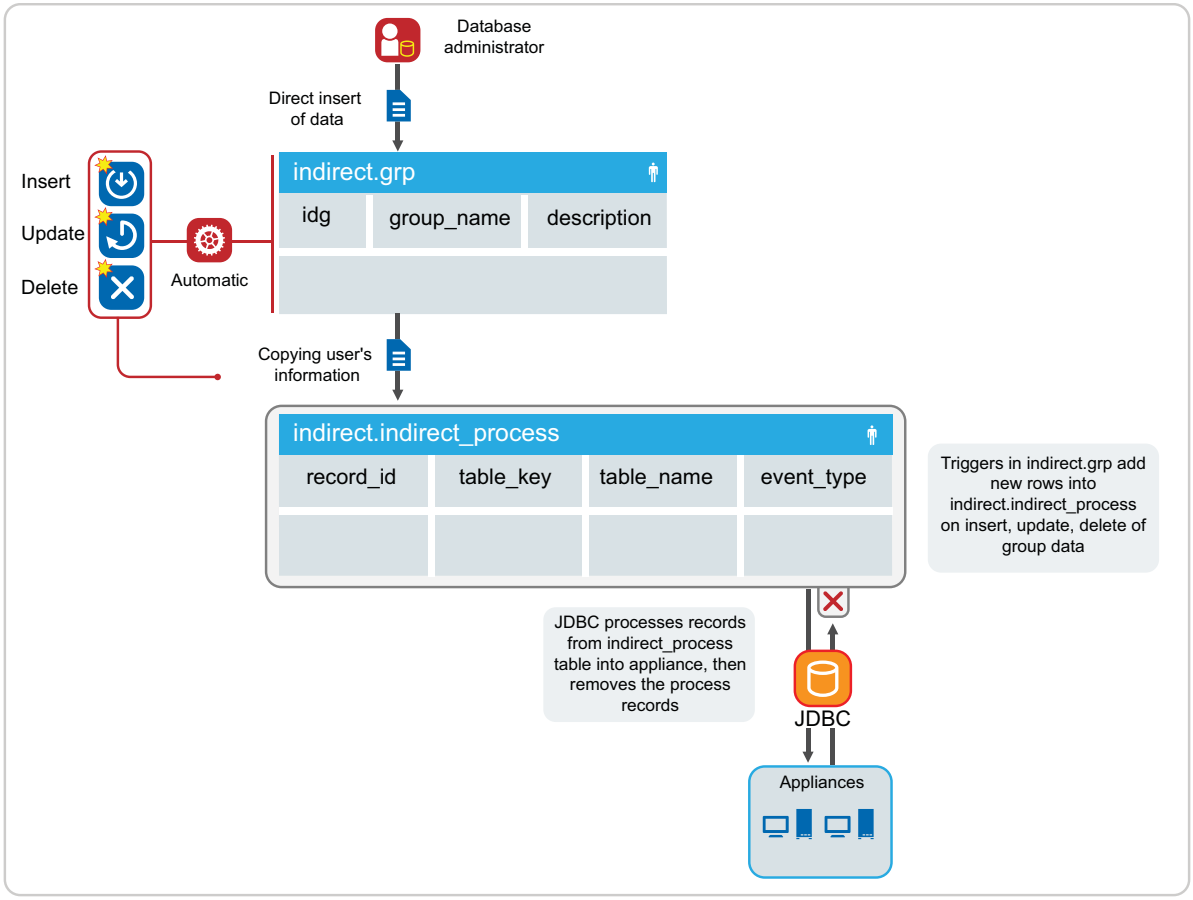
The database administrator creates the user accounts or logs in to the `user$` table. (The dataflow figures are based on the default Oracle security table `user$`.) The database administrator defines triggers or procedures that copy information into the `indirect.usr` table.

The `indirect_install` SQL script creates the automatic insert, update, or delete triggers on the `indirect.usr` table. When rows in the `indirect.usr` table are altered, the automatic triggers add a row to the `indirect.indirect_process` table.

The appliance polls the `indirect.indirect_process` table. When the appliance detects rows in the `indirect.indirect_process` table of type `user`, the appliance adds, modifies, or deletes the user account in the applications connected to the appliance.

The appliance then deletes the row from the `indirect.indirect_process` table after the appliance processes the information.

Figure 2-2 *Dataflow of Group Information*

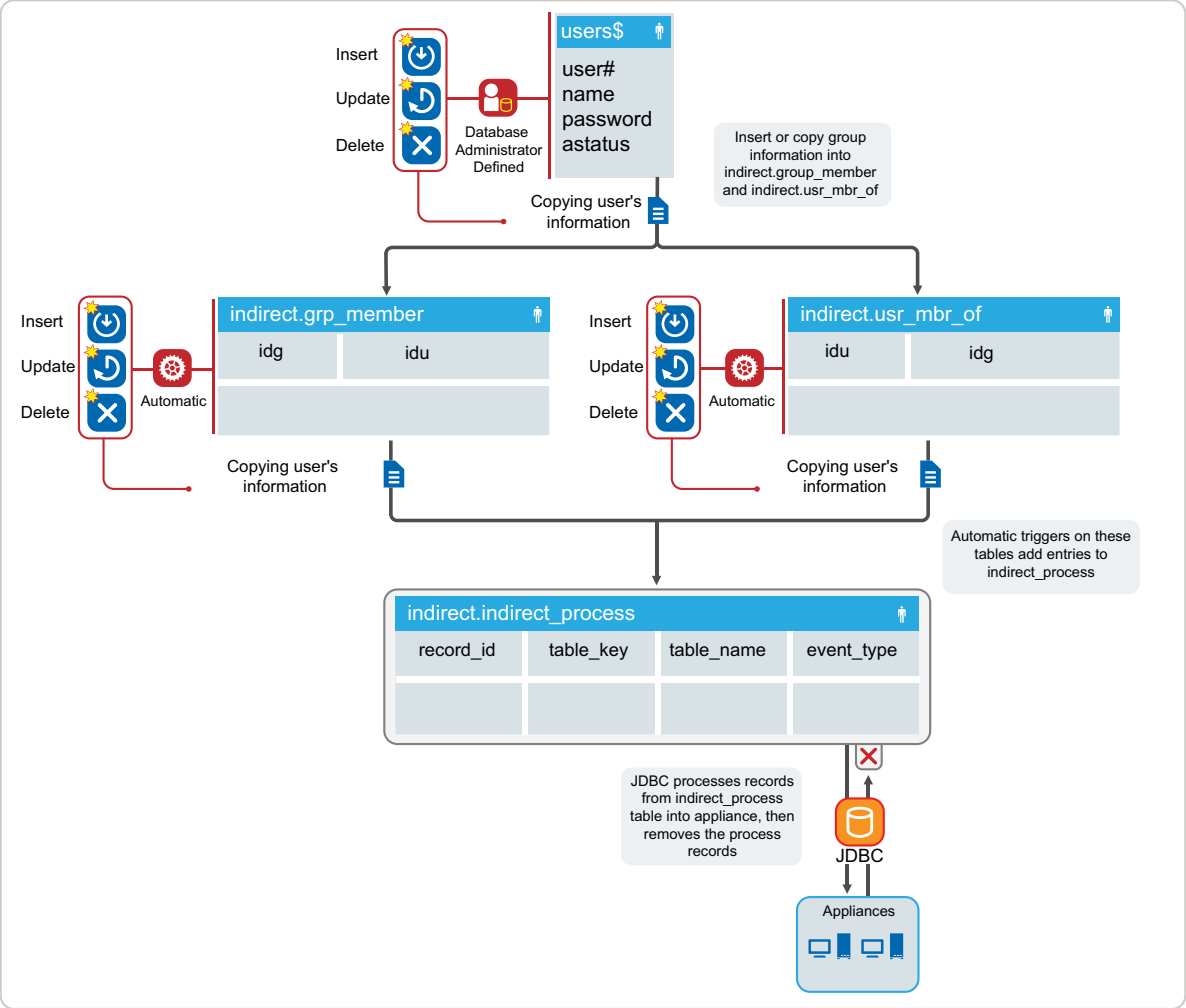


The database administrator performs a direct or triggered insert of data into the `indirect.grp` table.

The `indirect_install` SQL script creates the automatic insert, update, or delete triggers on the `indirect.grp` table. When rows in `indirect.grp` are altered, the automatic triggers add a row to the `indirect.indirect_process` table.

When the appliance detects rows in the `indirector.indirect_process` table of type group, the appliance adds, modifies, or deletes the groups in the applications connected to the appliance. The appliance then deletes the row from the `indirect.indirect_process` table after the appliance processes the information.

Figure 2-3 Relationship between Users and Groups



The indirect schema does not have a direct concept of group membership, but maintains a relationship between the `user idu` column and the `group idg` column in the tables `indirect.grp_member` and `indirect.usr_mbr_of` tables.

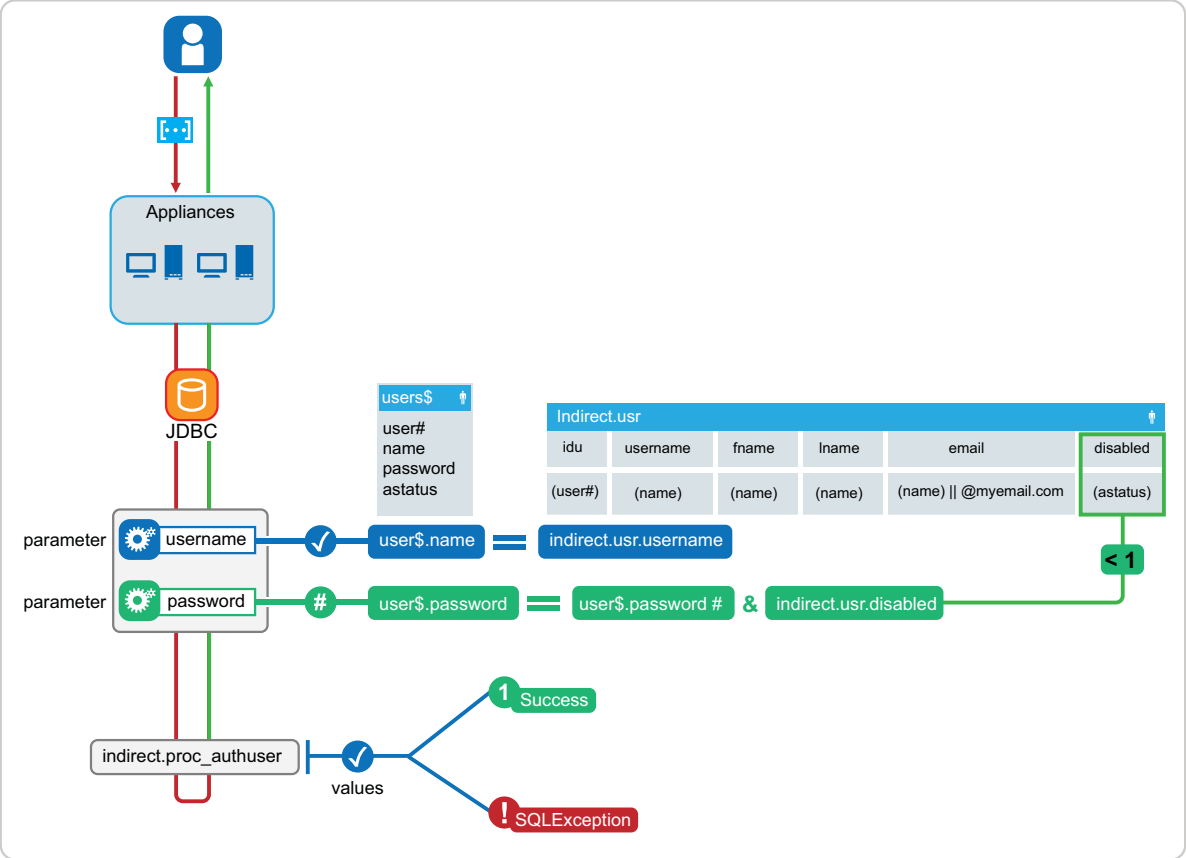
For group membership, the desired group `idg` and user `idu` must exist in both tables. The `indirect_install` SQL script creates the automatic insert, update, or delete triggers on the `indirect.grp_member` and `indirect.usr_mbr_of` tables. When rows in these tables are altered, the automatic triggers add a row to the `indirect.indirect_process` table.

When the appliance detects rows in the `indirect.indirect_process` table for group membership, the appliance adds, modifies, or deletes the group memberships in the applications connected to the appliance. The appliance then deletes the row from the `indirect.indirect_process` table after the connector processes the information.

For example, if the administrator wants to add a user with `idu 6` to a group with `idg 10`, they would have to manually (or through triggers) add entries into both the `grp_member` and `usr_mbr_of` tables.

```
"INSERT INTO indirect.grp_member(idg,idu) VALUES(10,6); INSERT INTO
indirect.usr_mbr_of(idu,idg) VALUES(6,10);"
```

Figure 2-4 *Authentication Process*



To verify authentication credentials, the appliance calls a stored procedure `indirect.proc_authuser` with the parameters of `@username,@password`. The procedure compares the `username` parameter with the default user table (`user$`) and the `indirect.usr.username` fields. If they matched, the process checks the `indirect.usr.disabled` flag (disabled > 0 = disabled). If login is enabled (disabled = 0), the process compares the `password` parameter to the existing password hash in the `user$` table. If the password hash matches, then the process authenticates the user successfully. If any of these conditions are not met, then the process returns an `SQLException` to the appliance, and authentication fails.

You can alter the stored procedure based on the desired schema that the database administrator wishes to use for authentication. The administrator needs to keep in mind that the stored procedure `indirect.proc_authuser`(`@username,@password`) is hard-coded into the appliance, and expects either a `success (1)` or `SQLException` returned.

2.4.4 Requirements for Other Identity Sources

The CloudAccess appliance allows you to use additional identity sources after you deploy the appliance. However, these additional identity sources cannot be used during the initialization of the appliance.

Requirements for Self-Service User Store

The Self-Service User Store is an internal identity source you can use with the appliance. However, you can use this identity source only after you configure the appliance. The Self-Service User Store cannot be used during the initialization process of the appliance.

There are no specific requirements to use this service. For more information, see Section 4.2.1, “Configuring Self-Service Registration and Password Management,” on page 42.

Requirements for SAML Federation Partner as Identity Source

You can create a custom connector that allows users to authenticate to the CloudAccess appliance through a SAML federated connection. In order to create this federation, you must create a custom SAML In connector.

For more information, see Section 7.13.6, “Creating a SAML 2.0 Inbound (SAML2 In) Connector Template,” on page 138.

2.5 Appliance Installation Worksheet

Use the following worksheet to gather the required information to install and configure the appliance.

Table 2-4 *Appliance Installation Worksheet*

Component	❑	Gather the following information:
Networking Information		
	❑	Publicly resolvable DNS name for the appliance
	❑	NTP server
	❑	DNS server, subnet mask, and gateway
	❑	(Recommended) An SSL certificate signed by a well-known certificate authority (CA)
Identity Sources		
	❑	IP address or DNS name of the server or database that contains the users
	❑	Context of the users (Only for Active Directory and eDirectory)
	❑	Name and password of a user with the proper rights to the users

2.6 Deploying the Appliance

Whether you are deploying CloudAccess or MobileAccess, the appliance is an Open Virtualization Format (OVF) virtual appliance. You must deploy the appliance to your VMware server.

To configure the appliance, the appliance must obtain an IP address through DHCP or have an assigned static IP address. NetIQ provides two different OVF files for each appliance, to accommodate DHCP and non-DHCP environments:

- **DHCP environment:** An `*.ovf` file for environments that have a DHCP server.
- **Non-DHCP environment:** A `*-vcenter.ovf` file for environments that do not have a DHCP server and need to use a static IP address.

To deploy the appliance:

1. Download the appropriate artifact from the NetIQ Downloads web page (https://dl.netiq.com/).
2. (Conditional) If you are using Windows, extract the VMware image to access the available OVF file.
3. (Conditional) If you are using Linux, use the following command to extract the image:

   ```
   tar -zxvf vmware_image.tar.gz
   ```

4. (Conditional) If you have a DHCP server in your environment, deploy the `*.ovf` file to a specific ESXi host. For more information, see the VMware documentation.
5. (Conditional) If you do not have a DHCP server in your environment:

   5a Deploy the `*-vcenter.ovf` file to a VMware vCenter Server, using either the command line tool `ovftool` or the VMware vSphere client.

   5b Configure the appliance properties, ensuring that you change the `use_dhcp` property to false. Other required properties include the static IP address, subnet mask, default gateway, DNS server, and NTP server name.

   TIP: If you deploy the appliance using the `ovftool`, you can configure the appliance properties from the command line and auto-start the VM so you do not have to use the vSphere client to configure the properties before starting the VM.

6. Power on the appliance, then proceed to Section 2.8, “Initializing the Appliance,” on page 27.

The initial boot configures the appliance. The initial boot could take between five and twenty minutes for the configuration to complete. When the appliance is ready, it displays a welcome message with the initialization URL `https://appliance_ip_address/appliance/Init.html`.

NOTE: Whether you have a MobileAccess-only license or a full CloudAccess license, you need to install only one virtual appliance to access all features. The CloudAccess appliance includes all MobileAccess features.

2.7 Upgrading Your Environment

If you are currently running CloudAccess 2.0 or 2.1 in your environment, you can use the update channel to upgrade to CloudAccess 2.2. For more information, see Section 11.5, “Updating the Appliance,” on page 164.

2.8 Initializing the Appliance

Once you have deployed the appliance, you must initialize it.

1. Verify that you meet the requirements listed in Section 2.2, “Product Requirements,” on page 14.
2. From a supported browser, access the initialization web interface at the URL displayed on the appliance screen after it is deployed.

   For example: `https://appliance_ip_address/appliance/Init.html`

   NOTE: This URL is case-sensitive, so ensure that you enter the non-variable portions of the URL exactly as illustrated.

3. Provide the following information needed to initialize the appliance:

   3a Initialize Appliance

   Select **Join Cluster** only if you are initializing an appliance to add to an existing cluster. The first appliance that you configure automatically becomes the master node in the cluster.

   3b Step 1 - Network

   NTP server - The NTP (Network Time Protocol) server allows appliances to synchronize time with each other. Without this server, the data on the appliances becomes corrupted.

   Obtain an IP address automatically - Select this option if you have a DHCP server to provide the IP address of the appliance.

   Use the following IP address - Select this option only if you do not have a DHCP server in your environment and you are assigning a static IP address. If you deployed the appliance using the `*-vcenter.ovf` file, the fields are auto-populated with the settings you specified during deployment.

   3c Step 2 - Identity Source

   Select the identity source that you plan to use.

   Username and **Password** - Specify the fully distinguished LDAP format name of a user in the identity source who has Read access to the identity source. For example, in eDirectory this might be `cn=admin,o=netiq`.

   Context - Specify the search context of the users in the identity source. For example, `o=netiq`.

   Enable LDAP SSL - Specify whether you want to use SSL for communication with an LDAP identity source. If you use SSL, the default port is 636. The default port for non-SSL is 389. You can specify other port numbers as required if the identity source is using a non-default port.

   When you set up the identity source for the first time, you specify a single replica of the identity source, but you can add more replicas later if needed.

   3d Step 3 - Cluster Information

   Public DNS - Specify the public DNS name that is used as the base URL to access the appliance. For example, `nca-01.company.info`.

   Admin user name - Specify the user account in the identity source search context who becomes the administrator of the appliance. In eDirectory, this is the CN of the user. In Active Directory, this is the sAMAccountName of the user.

3e **Step 4 - Appliance Password**

New password / Confirm password - Specify the password for the appliance administrator account. If you lose the connection to the identity source for any reason, you can run through these initialization steps again using the appliance password specified here. You would then specify a different identity source and the user name and password of the new appliance administrator in that new identity source.

4 Click **Finish**.

A successfully initialized appliance automatically redirects the browser to the administration console login page at `https://appliance_dns_name/appliance/index.html`.

5 Log in with the Admin user name specified in **Step 3 - Cluster Information**. The password is the user's password in the identity source.

6 Proceed with Chapter 3, "Configuring the Appliance," on page 29.

You can change the initialization settings at any time if needed. Enter `appliance_dns_or_IP_address/appliance/Init.html` in a browser to access the initialization settings page. After the appliance has been initialized for the first time, the next time you access the `Init.html` page, CloudAccess prompts you for the appliance password.

Whenever you make changes to the appliance, click **Apply** and wait for the appliance to finish applying your changes. Do not attempt to perform any other administration tasks in the console until the gears have stopped spinning on the appliance icon.

3 Configuring the Appliance

Once you have installed and initialized the appliance, configure the appliance to communicate with the SaaS applications.

3.1 Accessing the Administration Console

After you properly initialize the appliance using the information in Section 2.8, “Initializing the Appliance,” on page 27, the browser automatically redirects you to the administration console at `https://appliance_dns_name/appliance/index.html`. If the initialization does not automatically redirect you, open the page manually to complete the appliance configuration.

To access the administration console:

1. In a supported browser, enter `https://appliance_dns_name/appliance/index.html`.
2. Log in as the administrator of the appliance.

   These credentials are the cluster administrator user name specified during the initialization process and its identity source user password.
3. The first time you log in, the appliance displays the Admin page and might display user count activity on the **Users** bar as users in the search contexts from the identity source are imported and activated. Ensure that this process completes before you configure any application connectors.

Icons at the top of the Admin page allow you to access the other administration pages. If your session times out or you log out, the next time you log in to the appliance, CloudAccess displays the page that you last accessed.

Admin sessions time out by default after 10 minutes. This setting is not currently configurable, but you can adjust the timeout setting for user sessions. For more information, see Section 11.2, “Configuring Session Timeouts,” on page 163.

For a list of the different administration pages, see Section 3.2, “Getting Started,” on page 29.

3.2 Getting Started

From the Admin page you can use the navigation icons at the top of the page to access other functions in the administration console:

- **Roles:** Configure roles for different users within CloudAccess. For more information, see Section 3.6.2, “Assigning Roles to Users,” on page 35.
- **Policy:** Map roles (groups) from the identity source to authorizations from the SaaS applications. For more information, see Chapter 8, “Mapping Authorizations,” on page 151.
- **Approval:** Approve or deny authorizations for the SaaS applications. This icon appears only if you have mapped roles to authorizations and selected the option to require approval for accounts, and there are accounts waiting for approval. For more information, see Section 8.8, “Approving Requests,” on page 154.
- **Reports:** Report on the user activities to the SaaS applications. For more information, see Chapter 9, “Reporting,” on page 157.

- **Devices:** View and manage registered mobile devices. For more information, see Section 6.8, “Managing Mobile Devices,” on page 74.

Before you begin any configuration tasks, you should register your appliance. For more information, see Section 3.3, “Registering the Appliance,” on page 30.

3.3 Registering the Appliance

CloudAccess provides a 30-day trial period. If you do not register the appliance within 30 days after installation, the appliance stops working. The bomb icon on the Admin page displays how many days are left in the trial period.

For the purpose of meeting licensing requirements, when you register a single appliance, the cluster as a whole is considered to be registered. However, in order to use the Customer Center update channel to download and install software updates, you must register each node in the cluster separately. The bomb icon remains on the Admin page if there are nodes in the cluster that have not yet been registered for channel updates. For more information about the update channel, see Section 11.5, “Updating the Appliance,” on page 164.

To register your appliance:

1. Log in to your Customer Center at https://www.netiq.com/customercenter (https://www.netiq.com/customercenter).

   The Customer Center is for NetIQ, Novell, and SUSE customers.
2. Click **My Products** > **Products**, then click **CloudAccess**.
3. Click the right arrow on the line next to the product to open a details page.
4. Select the **Activation Code** value and copy it to the clipboard. You will need this code to register the appliance.
5. Log in to the appliance at `https://appliance_dns_name/appliance/index.html`.
6. On the Admin page, click the appliance, then click **Register appliance**.
7. Enter the email address you used when you registered with the Customer Center.
8. Paste the Activation Code you copied to the clipboard from the Customer Center.
9. Click **Register**.
10. Repeat Step 6 through Step 9 for each appliance in the cluster.

When you have successfully registered all nodes in the cluster, the bomb icon disappears.

3.4 Configuring Network Options

CloudAccess contains a manual routing table, supports two Network Interface Cards (NICs), and provides a forward proxy only for testing purposes.

- Section 3.4.1, “Configuring the Forward Proxy,” on page 31
- Section 3.4.2, “Configuring the Second Network Interface,” on page 31
- Section 3.4.3, “Configuring the Routing Table,” on page 32
- Section 3.4.4, “A Sample Network Configuration,” on page 32

3.4.1 Configuring the Forward Proxy

The forward proxy takes requests coming from the internal network and forwards these requests to the Internet.

The forward proxy feature has the following limitations:

- The forward proxy is intended only for testing purposes, and is not supported in a production environment.
- When forward proxy is enabled, Simple Proxy connectors work only with web servers or resources (as specified in the **Connects to** field of the connector configuration) that are on the local segment. All simple proxied services must be reachable from the appliance without going through the forward proxy.

  In addition, before you configure a Simple Proxy connector to work with a web server on the local segment, you must add the IP address of the web server to the Forward Proxy's **Ignore List** configuration field. Otherwise, the simple proxy configuration is not allowed, and the **Connects to** field turns red.

NOTE: Wildcard entries in the **Ignore List** field are not supported.

To configure the forward proxy:

1. Log in with an appliance administrator account to the Admin page at `https://`*`appliance_dns_name`*`/appliance/index.html`.
2. Drag and drop the **Forward Proxy** icon from the **Tools** palette to the **Tools** panel.
3. Use the following information to configure the forward proxy:

   Forward Proxy Server: Specify the IP address and port number for your proxy server.

   Ignore List: Specify any IP addresses with the associated DNS names that you want the forward proxy to ignore. For example, `127.0.0.0|localhost`.
4. Click **OK** to save your changes. Note that clicking **OK** causes the services to restart and you must log in to the appliance again.

3.4.2 Configuring the Second Network Interface

CloudAccess supports two Network Interface Cards (NICs) for each node in the cluster. You can configure one NIC for the administrative network and a second NIC for the public network. However, whether the nodes in a cluster each have one or two NICs configured, the cluster itself has only two DNS names: one for the Admin NICs and one for the Public NICs.

IMPORTANT: If you configure two NICs on one appliance, you must configure all other nodes in the cluster with two NICs.

To configure the second NIC on a node:

1. Log in with an appliance administrator account to the Admin page at `https://`*`appliance_dns_name`*`/appliance/index.html`.
2. Click a node icon, then click **Configure**.
3. Click the **Public Interface** tab.
4. Select **Enable Separate Public Interface**.
5. Configure the network settings for your public network and click **OK**.

6 (Conditional) If this is the first node in the cluster with a Public NIC, enter the DNS name for the public network. Modify the keypairs for SSL and SAML as needed and click **OK**.

7 Click **Apply** to save the changes.

8 Click **Close**.

9 Repeat Step 2 through Step 8 for each node in the cluster.

3.4.3 Configuring the Routing Table

CloudAccess provides a routing table for your use if your network has static routes. The routing table allows you to define the next hop in your network for the node in the cluster to reach the desired destination.

To configure the routing table for each node:

1 Log in with an appliance administrator account to the Admin page at `https://appliance_dns_name/appliance/index.html`.

2 Click the node icon, then select **Configure**.

3 Click the **Routing** tab.

4 Specify the appropriate **Reverse Path Filter** setting. Reverse path filtering is used to prevent packets that arrived through one interface from leaving through a different interface. If in doubt, leave the default setting of **Strict mode**, since it prevents users from spoofing IP addresses from local subnets and reduces the likelihood of distributed denial-of-service (DDoS) attacks.

5 Click the plus sign (+) icon to add a route.

6 Define the desired route, then click **OK**.

7 (Optional) Add additional routes.

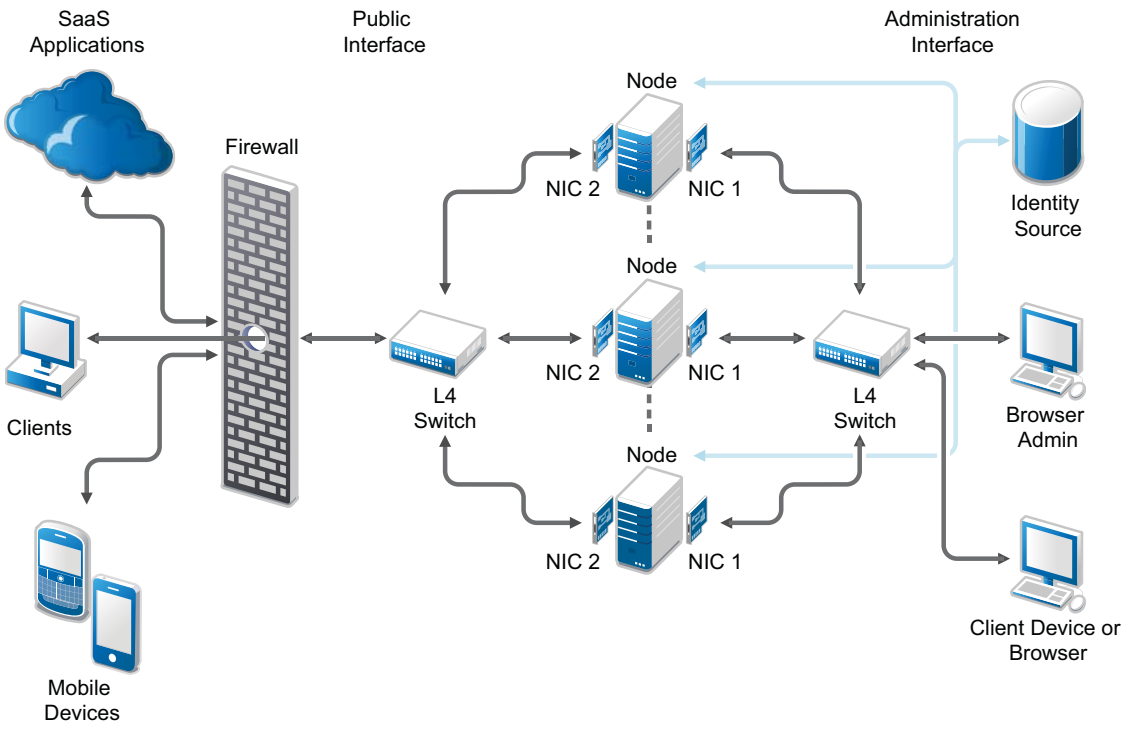
8 Click **Close**.

9 Repeat Step 2 through Step 8 for each node in the cluster.

3.4.4 A Sample Network Configuration

The following graphic depicts a possible network configuration using CloudAccess with both NICs enabled on each node.

Figure 3-1 *A Sample Network Diagram*



The network diagram shows that each node has both NICs enabled. The first NIC is the administration interface for the node and the second NIC is the public interface of the node. All of the administration and corporate information stays on the administration interface side of the network. All user requests and application requests communicate only on the public interface. This configuration provides a layer of security for your corporate information.

3.5 Changing the Certificates on the Appliance

The appliance contains SSL and SAML self-generated certificates, by default both named ag4csrv1, but NetIQ highly recommends that you replace the default certificates with well-known Certificate Authority signed certificates. The required format for importing a key pair is `.pfx`. This format contains the private key, certificate, and trusted roots required to import.

To change the certificates:

1. Log in with an appliance administrator account to the Admin page at `https://`*`appliance_dns_name`*`/appliance/index.html`.
2. Click the cluster icon under **Appliances**, then click **Configure**.
3. Delete the default key pairs by clicking the red delete (X) icon next to the SSL key pair and the SAML key pair.
4. Browse to and select the certificates you want to use, then click **OK**.
5. In the Instructions window, click **OK**.
6. Click **Apply** and wait for the configuration changes to be applied to the appliance. Do not perform other administration tasks in the console while the changes are being applied.
7. Close your browser and reopen it to start a new session using the new key pairs.

Expired key pair certificates prohibit changes from being made to this page and make the key pair field red. If the key pair expires, you must re-initialize the appliance before you can upload a new certificate.

3.6 Configuring Roles Management

CloudAccess provides the ability to assign different roles to administrative users in your identity sources. The roles allow administrators to perform certain tasks and deny them access to other tasks.

- Section 3.6.1, “Defining the Role Types,” on page 34
- Section 3.6.2, “Assigning Roles to Users,” on page 35

3.6.1 Defining the Role Types

CloudAccess includes the following types of roles:

- **Appliance Administrator:** One or more users can have the appliance administrator role in CloudAccess. You assign the first appliance administrator account during the initialization of the appliance.

  The first appliance administrator has rights to access the Admin, Roles, Policy, Reports, and Devices pages. This administrator has rights to add and remove identity sources, tools, nodes, and applications. However, this administrator does not have application ownership and approval rights by default.

  When another appliance administrator assigns a user the appliance administrator role, that new administrator also has rights to access the Admin, Roles, Policy, Reports, and Devices pages. However, like the first appliance administrator, subsequent administrators do not have ownership and approval rights by default.

  Only appliance administrators can add SaaS applications, such as Google Apps and Salesforce. Any appliance administrator that imports and applies a SaaS application is also made that application owner and has rights to the Approvals page for that specific application. An administrator who is also an application owner can assign another administrator to be the application owner and application approver, and can either keep or remove himself from those roles.
- **Application Owner:** The application owner controls access to the SaaS applications. CloudAccess automatically assigns this role to the user who creates the SaaS application on the Admin page. The application owner can access the following web pages:
  - **Approvals:** The application owner can allow or deny approvals for users to obtain a SaaS application account.
  - **Policy:** The application owner can map authorizations between the identity source and the SaaS application and optionally require approval for authorizations.
  - **Roles:** The application owner can add or remove users from the application approver role.
- **Application Approver:** The application approver can access the Approvals page and allow or deny approvals for users to obtain a SaaS application account. CloudAccess automatically assigns this role to the administrator who creates the SaaS application on the Admin page.
- **Compliance Auditor:** The compliance auditor can access the Reports page and generate, view, and download the reports for the appliance. Users assigned to the appliance administrator role automatically have access to the Reports page.

- **Device Administrator:** The device administrator can view and delete other users' registered mobile devices on the Devices page. Users assigned to the appliance administrator role automatically have the device administrator role (though the reverse is not the case).
- **Helpdesk:** The helpdesk administrator manages the Self-Service User Store users. The helpdesk user can delete users and reset passwords.

In addition to the default role assignments, you can assign each role to additional users. However, the Roles page never allows you to remove the last appliance administrator role.

3.6.2 Assigning Roles to Users

To assign roles to users:

1. Log in to the Admin page at `https://appliance_dns_name/appliance/index.html` as the appliance administrator or application owner.
2. Click **Roles** on the toolbar.
3. Type the name of a user into the search filter field, then click **Search**. Matching users are displayed in the left column.

   NOTE: If you use wildcards such as an asterisk (*) or question mark (?) in the search filter field, CloudAccess does not correctly filter results. Filters must be full regular expressions. If you want to use wildcards, they must be regular expression wildcards. If the filter does not start with '^' and '.*', then '.*' is added to the filter. If the filter does not end with '$' and '.*', then '.*' is added to the filter. Thus, a filter for "test" would end up as the regular expression ".*test.*".

4. Drag and drop the user to the role you want to assign to that user, then click **OK** to confirm the assignment.

The Roles page displays only the application owner and application approver roles of configured SaaS connectors.

3.7 Configuring Clustering

You can cluster the CloudAccess appliance. By default, it is a single node cluster, but CloudAccess supports up to a five-node cluster. You add a node to the cluster by selecting **Join Cluster** during the initialization process.

- Section 3.7.1, "Advantages of Clustering," on page 35
- Section 3.7.2, "Managing Nodes in the Cluster," on page 36
- Section 3.7.3, "Configuring an L4 Switch for Clustering," on page 38
- Section 3.7.4, "Configuring an L4 Switch for Email Proxy," on page 38

3.7.1 Advantages of Clustering

Clustering in CloudAccess offers several advantages. Most of these advantages are available only if you configure an L4 switch or Round-robin DNS. The L4 switch is the best solution.

Disaster Recovery: Adding additional nodes to the cluster provides disaster recovery for your appliance. If one node stops running or becomes corrupt, you can promote another node to master.

High Availability for Authentications: CloudAccess provides high availability for authentications and the single sign-on service, when using an L4 switch in conjunction with clustering. This solution allows users to authenticate in case of problems with the nodes within the cluster. The L4 switch sends authentication requests to the nodes with which it can communicate.

Load Balancing: You can configure the L4 switch to distribute authentications to nodes so one node does not receive all authentication requests while other nodes sit idle.

Scalability: Configuring an L4 switch with clustering increases the scalability of CloudAccess. Each node in the cluster increases the number of possible simultaneous logins.

3.7.2 Managing Nodes in the Cluster

CloudAccess supports up to five nodes in a cluster. You add nodes to the cluster through the initialization process, and perform all other initialization tasks on the Admin page.

- "Adding a Node to the Cluster" on page 36
- "Promoting a Node to Master" on page 36
- "Removing a Node from the Cluster" on page 37

Adding a Node to the Cluster

To add a node to the cluster:

1. Verify that the cluster is healthy.
   - All nodes must be running and communicating.
   - All components must be in a green state.
   - All failed nodes must be removed from the cluster.

   For more information about verifying that your cluster is healthy, see Section 12.4, "Troubleshooting Different States," on page 169.
2. Download and deploy a new virtual machine (VM) for the new node.

   For more information, see Section 2.6, "Deploying the Appliance," on page 26.
3. Initialize the appliance. Select **Join Cluster** as the first step to initialize the new node, then follow the on-screen prompts.

   For more information, see Section 2.8, "Initializing the Appliance," on page 27.

   When initialization is complete, the browser is redirected to `index.html` and a login page appears.
4. Log in to `index.html` and verify that the new appliance appears in the cluster. Wait until all spinner icons stop processing and all components are green before performing any other tasks.

   The cluster is adding the node and there are several background processes running. This final step could take up to an hour to complete.
5. Once the node is added to the cluster, register the node. For more information, see Section 3.3, "Registering the Appliance," on page 30.

Promoting a Node to Master

The first node that you install is the master node of the cluster by default. The master node runs provisioning, reporting, approvals, and policy mapping services. You can promote any node to become the master node.

To promote a node to master:

1. Verify that the cluster is healthy.
   - All nodes must be running and communicating.
   - All components must be in a green state.
   - All failed nodes must be removed from the cluster.

   For more information about verifying that your cluster is healthy, see Section 12.4, "Troubleshooting Different States," on page 169.
2. Verify that all nodes in the cluster are running the same version of CloudAccess. If any nodes need to be updated, ensure that you update the nodes *before* you switch the master node. For more information, see Section 11.5, "Updating the Appliance," on page 164.
3. Take a snapshot of the cluster.
4. Click the node to become the master node on the Admin page, then click **Promote to master**.

   An M appears on the front of the node icon indicating it is now the master node. This process may take a while to complete. Watch for the node spinner icons to stop and Health indicators to turn green before proceeding with any additional configuration changes.

The services move from the old master to the new master. The old master is now just a node in the cluster.

WARNING

- If the old master node is down when you promote another node to master, remove the old master from the cluster, then delete it from the VMware server. Otherwise, the appliance sees two master nodes and becomes corrupted.
- When you switch the master node, the logs start again on the new master and reports start again on the new master. The historical logs are lost. The reporting data is also lost, unless you are using Sentinel Log Manager. For more information, see Section 9.2, "Integrating with Sentinel Log Manager," on page 157.

Removing a Node from the Cluster

You can remove a node from the cluster if something is wrong with the node. However, ensure that you use the following steps to properly remove the node. If you simply delete a node from the cluster, the appliance deletes the node from the interface, but the VMware image still exists and continues to run. Leaving the VMware image running allows users to authenticate to a node that does not exist on the Admin page.

NOTE: After you remove a node, you cannot add the same VM instance back into the cluster. You must delete this instance of the appliance from your VMware server, then deploy another instance to the VMware server to add a node back into the cluster.

To remove a node from the cluster:

1. (Conditional) If the node you are removing is the master node, promote another node to be master before you remove the old node. For more information, see "Promoting a Node to Master" on page 36.
2. (Conditional) If you are using an L4 switch, delete the node from the L4 switch. For more information, see the L4 switch documentation.
3. On the Admin page, click the node you want to remove from the cluster.

4 Click **Remove from cluster**.

The Admin page immediately shows that the node is gone, but it takes some time for the background processes to finish.

5 Stop the VMware image on the VMware server, and then delete the instance of the node from the VMware server.

3.7.3 Configuring an L4 Switch for Clustering

If you want high availability or load balancing, you must configure an L4 switch for the CloudAccess appliance. An L4 switch can be configured in many different ways. Use the following recommendations to configure the L4 switch to work with the appliance.

- **Heartbeat:** Use the following URL to define the heartbeat for the L4 switch:

  `https://appliance_ip_address/osp/h/heartbeat`

  The L4 switch uses the heartbeat to determine if the nodes in the cluster are running and working properly. The heartbeat URL returns a text message of Success and a 200 response code.

- **Persistence:** Also known as **sticky sessions**, persistence allows all subsequent requests from a client to be sent to the same node. To make this happen, select SSL session ID persistence when configuring the L4 switch.

Session persistence ensures that the same real server is used for the CloudAccess login and the subsequent application single sign-on. Using the same server allows caching for a series of related transactions, which can improve the server performance and reduce the latency of transactions. It removes the delay that might occur if the client sends a request to a new node instead of using the existing session to the same node. To ensure that transactions for the same client are forwarded to the same real server in a load-balanced cluster configuration:

- You can set the L4 switch to use IP-based persistence, which uses the user device's IP address to maintain an affinity between the user session and the same real server in the cluster. IP-based persistence fails if a user's device IP address changes between requests, such as if a user's mobile device changes networks during a session. It also fails if all user devices come through a proxy service where all transactions appear to come from the same IP address.
- You can set the L4 switch to use sticky-bit persistence. Sticky-bit persistence is problematic for L4 switches that do not support stickiness. Sticky sessions also do not work with browsers set to disable cookies.
- You can use a proxy approach for the identity provider nodes that does not depend on the L4 configuration. However, this solution can quickly become chatty.

3.7.4 Configuring an L4 Switch for Email Proxy

CloudAccess contains an email proxy for users with Google Apps that supports three protocols: SMTP, POP3S, and IMAPS. You must configure your L4 switch to handle these protocols. Use the following high level steps to configure the protocols for your L4 switch. For more information, see your specific L4 documentation.

- "Configuring the SMTP Protocol Handler" on page 39
- "Configuring the POP Protocol Handler" on page 39
- "Configuring the IMAP Protocol Handler" on page 40

Configuring the SMTP Protocol Handler

To configure an SMTP protocol handler for your L4 switch:

1. On your L4 switch, configure a new IP group (traffic group) or use an existing group for the virtual servers in the L4 switch.

   You can use this group for all of the protocols.

2. (Optional) Create a health monitor:
   - **2a** Set the health checking for the pool to **TCP transaction monitor**.
   - **2b** Set the timeout to 30 seconds.
   - **2c** Set the health monitor to separately monitor each node.
3. Create a traffic pool for the SMTP virtual server to use:
   - **3a** Add each appliance node to the pool using the IP address with the port.

     For example: 192.168.1.14:25. The SMTP port is 25.
   - **3b** (Optional) Add the health monitor created in Step 2.
   - **3c** Select your load balancing settings.

     For example: round robin or random
   - **3d** Set the session persistence to **SSL Session ID**.
4. Create a new virtual server:
   - **4a** Specify the protocol as SMTP and the port as 25.
   - **4b** Use the traffic group defined in Step 1 and the pool defined in Step 3 for the virtual server.
5. Start the virtual server.

Configuring the POP Protocol Handler

To configure a POP protocol handler for your L4 switch:

1. On your L4 switch, configure a new IP group (traffic group) or use an existing group for the virtual servers in the L4 switch.

   You can use this group for all of the protocols.

2. (Optional) Create a health monitor:
   - **2a** Set the health checking for the pool to **TCP transaction monitor**.
   - **2b** Set the timeout to 30 seconds.
   - **2c** Set the health monitor to separately monitor each node.
3. Create a traffic pool for the POP virtual server to use:
   - **3a** Add each appliance node to the pool using the IP address with the port.

     For example: 192.168.1.14:995. The POP port is 995.
   - **3b** (Optional) Add the health monitor created in Step 2.
   - **3c** Select your load balancing settings.

     For example: round robin or random
   - **3d** Set the session persistence to **SSL Session ID**.

4 Create a new virtual server:

   4a Specify the protocol as SSL (POP3S) and the port as 995.

   4b Use the traffic group defined in Step 1 and the pool defined in Step 3 for the virtual server.

5 Start the virtual server.

Configuring the IMAP Protocol Handler

To configure an IMAP protocol handler for your L4 switch:

1 On your L4 switch, configure a new IP group (traffic group) or use an existing group for the virtual servers in the L4 switch.

   You can use this group for all of the protocols.

2 (Optional) Create a health monitor:

   2a Set the health checking for the pool to **Connect**.

   2b Set the health monitor to separately monitor each node.

3 Create a traffic pool for the IMAP virtual server to use:

   3a Add each appliance node to the pool using the IP address with the port.

      For example: 192.168.1.14:993. The IMAP port is 993.

   3b (Optional) Add the health monitor created in Step 2.

   3c Select your load balancing settings.

      For example: round robin or random.

   3d Set the session persistence to **SSL Session ID**.

4 Create a new virtual server:

   4a Specify the protocol as SSL (IMAPS) and the port as 993.

   4b Use the traffic group defined in Step 1 and the pool defined in Step 3 for the virtual server.

5 Start the virtual server.

3.8 Configuring CloudAccess to Forward Events to a Syslog Server

You can configure CloudAccess to forward various events to a syslog server. Event types that are forwarded include Login, Logout, Register Device, Un-register Device, and Failed Login.

To configure CloudAccess to forward events to a syslog server:

1 Log in with an appliance administrator account to the Admin page at `https://`*`appliance_dns_name`*`/appliance/index.html`.

2 Drag and drop the Syslog tool from the **Tools** palette to the **Tools** panel.

3 In the **Tools** panel, click the Syslog tool, then click **Configure**.

4 Specify the IP address and the port of the syslog server.

5 Select the type of protocol to use: **UDP**, **TCP**, or **TLS**.

6 Click **OK** to save the tool settings.

7 On the Admin page, click **Apply** to activate event forwarding.

4 Configuring Identity Sources

During the initialization of the CloudAccess appliance, you must configure one identity source. The appliance supports more than one identity source and there are additional identity sources you can add to the appliance that are not available during the initialization process.

IMPORTANT: You can add as many additional identity sources as you want. However, all of the user IDs across the different identity sources must be unique.

Use the information in the following sections to configure additional identity sources and verify that the user account information in the existing identity source contains the proper information to synchronize the accounts.

- Section 4.1, “Verifying the Identity Source User Attributes,” on page 41
- Section 4.2, “Configuring Additional Identity Sources,” on page 41

4.1 Verifying the Identity Source User Attributes

CloudAccess supports the use of one or more identity sources to authenticate users and for provisioning accounts to the SaaS applications. The initialization process configures the first identity source and adds the identity source to the Admin page.

To successfully provision users to the SaaS applications, you must ensure that each user account contains specific information for the appliance to provision the users. If you are using an LDAP identity source, there are specific attributes that must be populated on the user accounts. If you are using a JDBC database, there are certain columns of information that must be populated. The information for each identity source is different. For more information, see Section 2.4, “Identity Source Requirements,” on page 17.

For security reasons, by default CloudAccess does not allow you to add a user with a user name that is the same as a previously added user. If you attempt to do so, CloudAccess displays the user as not activated. For more information, see Section 12.6, “Troubleshooting Provisioning Issues,” on page 175.

4.2 Configuring Additional Identity Sources

During the initialization process, you configure an identity source. You can add more identity sources after the initialization process completes. However, the user IDs across the identity sources must be unique.

The Self-Service User Store (SSUS) and SAML 2.0 Inbound (SAML2 In) are identity sources that you can add only after you initialize the appliance. Use the following information to configure additional identity sources.

- Section 4.2.1, “Configuring Self-Service Registration and Password Management,” on page 42
- Section 4.2.2, “Configuring Additional Identity Sources,” on page 47
- Section 4.2.3, “Configuring SAML 2.0 Inbound Identity Sources,” on page 47

4.2.1 Configuring Self-Service Registration and Password Management

The Self-Service Registration and Password Management tool (SSRPM) allows you to empower users to register for services and to manage their credentials. It provides selected services from the NetIQ Self-Service Password Reset tool. The Self-Service User Store (SSUS) stores identity and credentials for self-registered user accounts. It is an additional identity source you can use with the appliance.

- "Enabling a Self-Service User Store (SSUS)" on page 42
- "Using SSUS as an Authentication Source for an Application" on page 43
- "Using the Self-Service User Registration and Password Management Services" on page 44
- "Providing Helpdesk Services for Self-Registered Users" on page 45

Enabling a Self-Service User Store (SSUS)

A Self-Service User Store provides self-service registration and password management services. You can enable and activate one Self-Service User Store. After you enable the service, the users can immediately begin to self-register on the SSUS Registration page. Self-registered users can then log in and access public applications from the landing page. Policies are required to allow the self-registered users to access private applications.

By default, SSUS requires new users to have a valid email account to create an SSUS account. Users must be able to receive and respond to a verification email.

You can also configure which service options to support for your SSUS users. The services include a helpdesk, new user, change password, and forgotten password.

To enable the SSUS service:

1. Log in with an appliance administrator account to the Admin page at `https://`*`appliance_dns_name`*`/appliance/index.html`.
2. In the administration console, drag the **Self-Service User Store** icon from the **Identity Sources** palette and drop it in the **Identity Sources** panel.
3. In the **Identity Sources** panel, click the new identity source, then click **Configure**.
4. On the **Configuration** tab, decide what options you want presented to your users and helpdesk administrators.
5. Click **OK** to enable the Service.
6. In the **System Configuration** panel of the administration console, click **Apply** to activate and start SSUS as a service.
7. Wait for the SSUS service to be activated and started across all nodes in the cluster.

   In the **Appliances** panel, the icon on each node of the cluster spins until the service is ready on the node. Do not apply additional changes until this action is complete on all nodes.

   A round green status icon in the lower left corner of the SSUS service icon indicates that the SSUS is configured and its status is healthy.

To allow the self-registered users to access to a private application that is enabled for SSUS, continue with "Using SSUS as an Authentication Source for an Application" on page 43.

Using SSUS as an Authentication Source for an Application

The applications are not available for SSUS users until you configure policies that authorize SSUS to be an authentication source for them. All SSUS users receive rights to an application when you assign a policy to the SSUS identity source.

- "Requirements for SSUS Authorizations" on page 43
- "Creating an SSUS Policy for a Private Application" on page 43
- "Deleting an SSUS Policy for an Application" on page 44

Requirements for SSUS Authorizations

Provisioning is not supported for users in an SSUS identity source. For more information, see "Requirements for Provisioning" in the *NetIQ® CloudAccess Connectors Guide*.

Creating an SSUS Policy for a Private Application

An application can have one or more authorizations for its resources. Each authorization can have one or more appmarks associated with it. You grant access to a private application by mapping one or more of its authorizations to the SSUS role. Users can access all of the appmarks associated with an authorization. You cannot control access at the appmark level.

NOTE: You should map authorizations for SSUS roles (groups) only to single sign-on applications. Do not map them to the SaaS applications with account provisioning (Google Apps, Office 365, or Salesforce).

To create a policy that grants access to an application for SSUS users:

1. Log in with an appliance administrator account to the Admin page at `https://`*`appliance_dns_name`*`/appliance/index.html`.
2. Click **Policy** in the toolbar.
3. On the Policy Mapping page, select **Other Identity Sources** from the drop-down list on the left, then select the **All SSUS Users** role.
4. On the right, select the software-as-a-service application that you want to use for this policy, and then view its authorizations.

   Some applications have multiple authorization options.
5. Drag the **All SSUS Users** role from the left side and drop it on the desired authorization on the right.

   You can also select multiple authorizations under a single application, then drag them from the right and drop them on the **All SSUS Users** role on the left.
6. In the pop-up **Mapping** window, review the mapped settings, then click **OK** to accept the new policy, or click **Cancel** to back out of the setup.

   You can remove an authorization in the list by selecting it, then clicking the **Delete** icon. A strike-through line is drawn through the entry.
7. Under **Other Identity Sources**, view the **Authorization** column for the **All SSUS Users** role to confirm that the **Authorization Stamp** icon appears.
8. Under the application on the right, view the **Policy** column to confirm that the **Policy** icon appears for the mapped authorization.

9 The appmarks for each of the mapped authorizations are available to users at their next login.

10 Repeat Step 4 through Step 9 for each application that you want to use SSUS as an identity source.

Deleting an SSUS Policy for an Application

You can deny access to an application by deleting its SSUS policy. Deleting the policy does not interrupt current sessions. The application is not available to users at their next login.

1 Log in with an appliance administrator account to the Admin page at `https://appliance_dns_name/appliance/index.html`.

2 Click **Policy** in the toolbar.

3 On the Policy Mapping page, select the **Self-Service User Store** role from the drop-down list on the left, then select the **All SSUS Users** role.

4 In the **All Users** row, click the **Authorization Stamp** icon.

5 In the **Edit All Users Mappings** window, select the authorization you want to remove, then click the **Delete** icon. Repeat this action for every authorization that you want to remove.

   A strike-through line is drawn through the entry.

6 Click **OK** to accept the modified settings, or click **Cancel** to back out of the setup.

   The application is not available to users at their next login.

Using the Self-Service User Registration and Password Management Services

The users can now self-register for accounts and manage their own credentials. Self-registered users can log in to the landing page for the appliance to access applications.

If you enabled the options during the SSUS configuration, users now see links on the appliance login page to create a new account or a link to reset their password if they have forgotten the password.

The user experience is:

1. The new user accesses the appliance login page.

   `https://appliance_dns_name`

2. The user clicks the link to create a new account.
3. The user follows the on screen prompts to create a new account that includes their name, email address, and a password.
4. After the account is created, the user is prompted to create security questions and answers.

   If the user forgets the account password, the questions and responses are used to verify the user's identity and allow the user to reset the password

5. (Conditional) If the user does not create the security questions and answers now, the user will be prompted to create them when they log in for the first time. If the user does not set up the security questions and answers the appliance will not authenticate the user. The user must set up the security questions and answers to authenticate to the appliance.
6. After the user completes the new account setup, the appliance sends a verification email to the user's email address. The user responds to verify the account creation.
7. The user accesses the login page again.

8. The user logs in with their new user name and password.
9. The user sees and can access the applications that the policies entitle them to see.

After you enable the Self-Service User Store, the Self-Service User Store login page is available for users. On this page, the user can perform the following tasks:

- **Create a New User:** A user can register for the service as a new user. The user provides their name and a valid email address, creates a password, and sets up security questions and responses. After the user responds to a validation message sent to the user's email address, the user can log in and begin using the account.
- **Change Password:** A self-registered user can reset the password for their account at any time. On the user's landing page, there is a **Change Password** icon they click. The user follows the on screen prompts to change their password.
- **Forgotten Password:** If a user forgets their password, they can reset their password after answering the security questions. The user access the login page and they enter their user name. The user then clicks the **Forgotten Password** link and follows the on screen prompts to change their password. The user must successfully reply to the security questions that the user set up for the account.

Providing Helpdesk Services for Self-Registered Users

The Self-Service User Store provides a helpdesk service. If a password expires or a self-registered user is locked out of an account, the password can be reset by an authorized helpdesk user for the SSUS service. The helpdesk user sets a temporary randomized password for the account, and the user is notified of the temporary password by email. This allows the user to log in to the account and reset the temporary password to use a custom password.

- "Adding a User to the Helpdesk Role" on page 45
- "Accessing the SSUS Helpdesk" on page 46
- "Resetting the Password for a Locked Self-Registered User Account" on page 46
- "Deleting a Self-Registered User Account" on page 46

Adding a User to the Helpdesk Role

You must assign a user to the SSUS Helpdesk role to provide helpdesk services for the related self-registered users.

NOTE: You should assign a user from an Identity source other than the SSUS source as the helpdesk user.

Use the following steps to enable the SSUS Helpdesk service.

1. The new user accesses the appliance login page at:

   `https://appliance_dns_name`

2. Click **Roles** in the toolbar.
3. On the Roles page, type the name of a user you want to assign to the Helpdesk role, click **Search**, then select the name.
4. Drag the user name from the left side and drop it on the Helpdesk role on the right.
5. In the pop-up **Add User to Role** window, review the mapped settings, then click **OK** to accept the new role assignment, or click **Cancel** to back out of the setup.

6 After the page refreshes, view the **Role** column for the user to confirm that the **Role** icon appears.
7 Under the Helpdesk role on the right, verify that the authorized user's name appears.

The helpdesk user accesses the helpdesk tools through the landing page. The helpdesk user logs in, and then clicks the **Helpdesk** icon on the landing page.

Accessing the SSUS Helpdesk

The landing page of an authorized helpdesk user displays a **Helpdesk** icon.

To access the Helpdesk from the landing page if you are an authorized helpdesk user:

1 Log in to CloudAccess using your corporate credentials.
2 On the landing page, click the **Helpdesk** icon to go to the Helpdesk page.

Resetting the Password for a Locked Self-Registered User Account

An authorized helpdesk user can use the SSUS Helpdesk service to reset the password for a self-registered user account. Typically, the user needs helpdesk assistance because the account is locked. If the account is not locked, the user can alternatively reset the password by using the **Forgotten Password** option on the Self-Service Registration login page.

To reset the password for an SSUS account as the authorized helpdesk user:

1 Log in to the SSUS Helpdesk as an authorized helpdesk user.
2 On the Helpdesk page, search for an SSUS user account, then click the self-registered user's name.
3 On the Password Policy page, view the password policy settings for the user account, then click **Change Password**.
4 On the Account Information page, confirm the user's information, then click **Change Password**.
5 In the **Random Passwords** window, select a password from the list of randomly generated passwords that satisfies the password policy for this account.

   You can click **More** to choose from additional random passwords.
6 View the confirmation message with the new password, then click **OK**.
7 After the self-registered user receives the temporary password, the user is prompted to reset the password at their next login.

Deleting a Self-Registered User Account

An authorized helpdesk user can use the SSUS Helpdesk service to delete the SSUS account for a self-registered user.

To delete an SSUS account as the authorized helpdesk user:

1 Log in to the SSUS Helpdesk as an authorized helpdesk user.
2 On the Helpdesk page, search for an SSUS user account, then click the self-registered user's name.
3 On the Account Information page, confirm the user's information, then click **Delete Account**.
4 Click **OK** to confirm the account deletion.

4.2.2 Configuring Additional Identity Sources

CloudAccess supports multiple types of identity sources. You can have one or more of each type of identity source configured on your appliance, and you can configure as many identity sources as you need.

The only restrictions are as follows:

- The source for each identity source must be unique. For example, do not configure multiple instances of an identity source for the same Active Directory domain.
- Every user account across the different identity sources must be unique.

For information about requirements and information that are needed to configure an Active Directory, eDirectory, or JDBC database identity source, see Section 2.4, “Identity Source Requirements,” on page 17.

NOTE: Although CloudAccess allows you to modify an existing eDirectory or Active Directory connector to point to a different tree, NetIQ does not recommend this approach because it can result in inconsistent display of user and group data. If you want to point a connector to a different tree, delete the existing connector and create a new connector that points to the correct tree.

To change the initial identity source configuration information:

1. Log in with an appliance administrator account to the Admin page at

   `https://appliance_dns_name/appliance/index.html`

2. In the **Identity Sources** panel, click the icon of the identity source that you want to modify, then click **Configure**.
3. Modify the settings as needed, then click **OK** to save the configuration information.
4. Click **Apply** to commit the changes to the appliance.

To add another identity source:

1. Log in with an appliance administrator account to the Admin page at

   `https://appliance_dns_name/appliance/index.html`

2. Drag and drop an identity source icon from the **Identity Sources** palette to the **Identity Sources** panel.
3. Click the identity source icon, then click **Configure**.
4. Complete the fields to configure the new identity source.
5. Click **OK** to save the configuration information.
6. Click **Apply** to commit the changes to the appliance.

4.2.3 Configuring SAML 2.0 Inbound Identity Sources

To allow the appliance to be a SAML 2.0 service provider, you can create a SAML 2.0 Inbound connector using the Access Connector Toolkit. SAML2 Inbound as an identity source is not available during initialization of the appliance. However, after you export the connector and import it in the appliance, the SAML2 In connector appears as an identity source. You configure an instance of the identity source with information about an appropriate identity provider in order to enable the service provider functionality of the appliance, and to allow the identity provider to send a SAML token to the appliance using the SAML 2.0 POST profile.

After you configure the SAML2 In identity source, the appliance login page provides a link to the login page of the SAML 2.0 identity provider, located to the left of the user name and password login options. The SAML 2.0 users log in through the identity provider to gain access to the appliance landing page.

For more information, see Section 7.13.6, “Creating a SAML 2.0 Inbound (SAML2 In) Connector Template,” on page 138.

5 Configuring Authentication Methods

The CloudAccess appliance allows you to configure one or more authentication methods for users. Some of the authentication methods are additive. For example, you can use Google reCaptcha with Integrated Windows Authentication. Use the information in the following sections to configure authentication methods for the appliance.

- Section 5.1, “Configuring Integrated Windows Authentication with Kerberos,” on page 49
- Section 5.2, “Configuring Google reCAPTCHA,” on page 51
- Section 5.3, “Configuring the Time-Based One-Time Password (TOTP) Tool for Two-Factor Authentication Using Google Authenticator,” on page 54
- Section 5.4, “Configuring the Advanced Authentication Tool for Two-Factor Authentication Using NetIQ Advanced Authentication Framework,” on page 61
- Section 5.5, “Configuring FIDO for Two-Factor Authentication,” on page 63
- Section 5.6, “Configuring the Authentication Filter to Set Session-Based Identity Information for a User,” on page 64

5.1 Configuring Integrated Windows Authentication with Kerberos

CloudAccess allows user authentication with either name and password or Integrated Windows Authentication with Kerberos if your identity source is Active Directory. If you choose to use Integrated Windows Authentication, you must configure Kerberos.

CloudAccess supports the use of only one Kerberos realm. If there are multiple Active Directory domains used as the identity source, all of the domains must use the same realm.

Use the information in the following sections to enable Kerberos authentication between Active Directory and CloudAccess.

- Section 5.1.1, “Configuring the Kerberos User in Active Directory,” on page 49
- Section 5.1.2, “Configuring the Appliance to Use Integrated Windows Authentication with Kerberos,” on page 50
- Section 5.1.3, “Configuring User Browsers,” on page 51

5.1.1 Configuring the Kerberos User in Active Directory

To configure Kerberos on your Active Directory domain:

1. As an Administrator in Active Directory, use MMC to create a new user within the search context specified during the initialization of the appliance.

   Name the new user according to the Host and DNS name of the appliance. For example, if the public DNS of the appliance is serv1.cloudaccess.com and the context that has been enabled for cloud is ou=acme corporation,dc=cloudaccess,dc=com, use the following information to create the user:

   First name: serv1

User login name: HTTP/serv1.cloudaccess.com

Pre-windows logon name: serv1

Set password: Specify the desired password. For example: Passw0rd

Password never expires: Select this option.

2 Associate the new user with the service principal name.

Any domain or realm references must be uppercase.

2a On the Active Directory server, open a cmd shell.

2b At the command prompt, enter the following:

`setspn -A HTTP/appliance`*`publicdns@UPN.SUFFIX newusershortname`*

For example: `setspn -A HTTP/serv1.cloudaccess.com@CLOUDACCESS.COM serv1`

2c Verify setspn by entering `setspn -L` *`shortusername`*

For example: `setspn -L serv1`

3 Generate the `keytab` file using the `ktpass` utility.

Any domain or realm references must be uppercase.

3a At the command prompt, enter the following:

`ktpass /out` *`filename`* `/princ` *`servicePrincipalName`* `/mapuser` *`userPrincipalName`* `/pass` *`userPassword`*

For example: `ktpass /out nidp.keytab /princ HTTP/serv1.cloudaccess.com@CLOUDACCESS.COM /mapuser serv1@CLOUDACCESS.COM /pass Passw0rd`

3b Ignore the message `Warning: pType and account type do not match`.

4 Copy the `nidp.keytab` file created in Step 3 to the browser of the client computer that you are using for administration.

5.1.2 Configuring the Appliance to Use Integrated Windows Authentication with Kerberos

The following steps enable the appliance to use Integrated Windows Authentication (IWA) with Kerberos, if your identity source is Active Directory.

1 Log in with an appliance administrator account to the Admin page at `https://`*`appliance_dns_name`*`/appliance/index.html`.

2 Click the **Active Directory** icon in the **Identity Sources** palette, then click **Configure**. (Do not drag the icon to the **Identity Sources** panel as you would when configuring the connector for Active Directory itself. These IWA configuration options are global for all connectors for Active Directory.)

3 Select **Integrated Windows Authentication**.

4 Next to the **Keytab** field click **Browse**, then browse to and select the `nidp.keytab` file generated in "Configuring the Kerberos User in Active Directory" on page 49.

5 Click **OK** to save the changes.

6 Click **Apply** to apply the changes to the appliance.

5.1.3 Configuring User Browsers

To complete the Kerberos configuration for Active Directory, configure the user browser. For more information, see Section 10.2, “Configuring End User Browsers for Kerberos Authentication,” on page 159.

5.2 Configuring Google reCAPTCHA

The Google reCAPTCHA tool helps protect your user login page against spam, malicious registrations, and other forms of attack where computers disguise themselves as human. It provides an additional layer of security by displaying images of words that users must type in addition to their login credentials. Software bots typically cannot scan the images to provide a response.

Using reCAPTCHA helps prevent automated Denial of Service (DoS) attacks that can impact the performance of the appliance and the identity source. The tool uses the remote Google reCAPTCHA service to provide the images and verify the responses. If a response succeeds, the appliance verifies the user’s authentication credentials against the identity source. If a response fails, the appliance fails the login attempt without processing the credentials, and re-displays the login page. Thus, the automated login attempts fail and cannot consume the processing resources of the appliance and identity source.

Use the information in the following sections to configure your system for reCAPTCHA:

- Section 5.2.1, “Requirements for reCAPTCHA,” on page 51
- Section 5.2.2, “Configuring Intrusion Detection for Failed Logins,” on page 52
- Section 5.2.3, “Configuring a Google reCAPTCHA Account,” on page 53
- Section 5.2.4, “Configuring the reCAPTCHA Tool,” on page 53

5.2.1 Requirements for reCAPTCHA

Ensure that your system meets the following requirements before you configure the Google reCAPTCHA tool:

- ❒ A CloudAccess appliance, installed and configured.
- ❒ One or more supported identity sources, with the connectors enabled and configured.

  The reCAPTCHA tool supports users from Active Directory, eDirectory, and Self-Service User Store (SSUS) identity sources. It does not support users from other types of identity sources, such as users imported from Microsoft SQL Server or Oracle Database type identity sources that use the JDBC identity source connector.

  Each identity source should be configured with an intrusion detection policy. For more information, see Section 5.2.2, “Configuring Intrusion Detection for Failed Logins,” on page 52.
- ❒ A Google reCAPTCHA account, configured on the Google reCAPTCHA website. For more information, see Section 5.2.3, “Configuring a Google reCAPTCHA Account,” on page 53.

5.2.2 Configuring Intrusion Detection for Failed Logins

Someone who attempts to use more than a few unsuccessful passwords while trying to log on to your system might be a malicious user. reCAPTCHA cannot prevent attacks by anyone who can read the image. It cannot differentiate between malicious users and legitimate users. Using reCAPTCHA cannot prevent coordinated human DoS attacks. If users have unlimited attempts to enter their authentication credentials, reCAPTCHA also cannot help prevent attacks to find passwords.

To help limit the effectiveness of brute force or human attacks that bypass the reCAPTCHA protection, you should enable the user's identity source to respond to this type of potential attack by disabling the account for a preset period of time after a specified number of failed logon attempts.

The supported identity sources have the following built-in intrusion detection systems:

- **Active Directory Account Lockout Policy:** Active Directory allows you to specify an account lockout policy for users and global security groups in a domain. Set the policy on the domain group policy object from the domain controller.

  To configure the Account Lockout Policy settings:

  1. Log in as an Active Directory administrator user to the Windows Server that hosts Active Directory Domain Services (the domain controller).
  2. Configure the Account Lockout Policy on the group policy object for the domain controller.

     For more information, see the *Account Lockout Policy* (http://technet.microsoft.com/en-us/library/hh994563%28v=ws.10%29.aspx) in the Microsoft TechNet Library. (http://technet.microsoft.com/)
  3. Verify that the **Account Lockout Threshold** value is higher than the number of failed login attempts you plan to specify for **Start reCAPTCHA at** in the reCAPTCHA tool.
  4. Repeat these steps for each configured Active Directory identity source.

- **eDirectory Intruder Lockout Policy:** eDirectory allows you to enable Intruder Detection and specify an Intruder Lockout policy for the container object where your user objects reside.

  To configure the eDirectory Intruder Detection and Intruder Lockout Policy:

  1. Log in as the eDirectory administrator user to the management console for the eDirectory server.
  2. Configure Intruder Detection and the Intruder Lockout policy on the container object where your user objects reside.

     For more information, see "Setting Up Intruder Detection for All Users in a Container" (https://www.netiq.com/documentation/edir88/edir88/data/afxkmdi.html#a3p5g0i) in the *eDirectory 8.8 SP8 Administration Guide*.
  3. Verify that the Intruder Lockout value is higher than the number of failed login attempts you plan to specify for **Start reCAPTCHA at** in the reCAPTCHA tool.
  4. Repeat these steps for each configured eDirectory identity source.

- **SSUS Lock Account After Detection:** The SSUS identity store automatically enables the Lock Account After Detection option. It allows up to 7 consecutive failed login attempts within a 30-minute interval. If the next login attempt also fails within the interval, SSUS locks the account for 15 minutes. After 15 minutes, the system automatically unlocks the account, and the user can log in using a correct user name and password. To log in before the lockout is reset, the user can contact the SSUS Helpdesk and ask the administrator to reset the password.

After you have configured intrusion detection for the supported identity sources, continue with Section 5.2.3, "Configuring a Google reCAPTCHA Account," on page 53.

5.2.3 Configuring a Google reCAPTCHA Account

Before you configure the Google reCAPTCHA tool, you must configure an account to use for your domain at Google reCAPTCHA, and create a public and private key.

To configure a Google reCAPTCHA account to use for your appliance's domain:

1. Access the Google reCAPTCHA (https://www.google.com/recaptcha/) website.
2. Click **Get reCAPTCHA > Sign up Now**.
3. Log in using one of your Google accounts.

   For example, if you use your Gmail account, the reCAPTCHA account is associated with the Gmail account.
4. (Conditional) If this is not your first site, click **Add a New Site**. Otherwise, skip to the next step.
5. Specify a domain.

   Read the **Tips** for more information.
6. Click **Create** to add the domain.
7. Copy the **Public Key** and **Private Key** that the interface displays to use when you configure the identity source.
8. Continue with Section 5.2.4, "Configuring the reCAPTCHA Tool," on page 53.

5.2.4 Configuring the reCAPTCHA Tool

Before you configure the Google reCAPTCHA tool, you must set up intruder detection in the Active Directory and eDirectory identity sources, and create public and private keys for your appliance's domain at the Google reCAPTCHA website.

To configure the reCAPTCHA service:

1. Using the identity source's native management tools, verify that their intrusion detection setup meets the requirements specified in "Configuring Intrusion Detection for Failed Logins" on page 52.
2. Log in with an appliance administrator account to the Admin page at

   `https://appliance_dns_name/appliance/index.html`
3. In the Identity Sources panel, verify that you have configured an identity source for Active Directory or eDirectory, or both.
4. Drag and drop the reCAPTCHA tool from the **Tools** palette to the **Tools** panel.

   The Configuration window opens automatically for the initial configuration. To view or reconfigure the settings later, click the reCAPTCHA tool, then click **Configure**.
5. Configure the reCAPTCHA feature as follows:

   Start reCAPTCHA at: Specify how many failed login attempts must occur before the login page displays the reCAPTCHA prompt. The value should be less than the lockout value set in the identity sources' intrusion detection system.

   - If the reCAPTCHA count is set to zero, the login page displays a reCAPTCHA prompt every time for all users. Every login requires user credentials and the reCAPTCHA response.
   - If the reCAPTCHA count is greater than zero, the login page displays the reCAPTCHA prompt only after the user login fails the specified number of times in the same browser window.

Public Key: Paste the Public Key value from your reCAPTCHA account configuration for this appliance's domain.

Private Key: Paste the Private Key value from your reCAPTCHA account configuration for this appliance's domain.

For information about the public and private keys for your reCAPTCHA account, see Section 5.2.3, "Configuring a Google reCAPTCHA Account," on page 53.

6 Click **OK** to save the settings and enable the tool.

7 Click **Apply** to activate the configuration.

8 Wait while the service is activated across all nodes in the cluster. Do not attempt other configuration actions until the activation completes successfully.

5.3 Configuring the Time-Based One-Time Password (TOTP) Tool for Two-Factor Authentication Using Google Authenticator

The Time-Based One-Time Password (TOTP) tool in CloudAccess supports the use of one-time passwords (OTPs) for two-factor authentication of users as they access applications through CloudAccess. With two-factor authentication, users must provide two categories of authentication factors before they can access the applications. The authentication factors used by the TOTP tool are:

- **Something the user knows:** The first authentication factor requires *something the user knows*, such as the password for the user's single-sign-on user name.
- **Something the user has:** The second authentication factor requires *something the user has*, such as a mobile device running Google Authenticator to generate time-based one-time passwords.

  Google Authenticator is a free software-token app that users deploy on their mobile devices. Authenticator generates time-based OTPs for authentication, without requiring an Internet connection or cellular service.

If users construct strong passwords and protect them, one-factor authentication can be an effective measure against security breaches. Two-factor authentication provides an additional layer of security to help ensure the identity of a user and reduce the risk of unauthorized access to your applications and data. Users still enjoy the convenience of single sign-on, but the access is more secure.

Figure 5-1 *The Two-Factor Authentication Solution for CloudAccess*

The following sections describe how to set up and use TOTP for CloudAccess:

- Section 5.3.1, “Understanding One-Time Passwords,” on page 55
- Section 5.3.2, “How to Use Google Authenticator for TOTP,” on page 56
- Section 5.3.3, “Configuring the TOTP Tool,” on page 57
- Section 5.3.4, “Registering a Mobile Device with the TOTP Tool for OTP Generation,” on page 58
- Section 5.3.5, “Using Two-Factor Authentication at Login,” on page 60
- Section 5.3.6, “Resetting a Device (Deregistering a Device),” on page 60

5.3.1 Understanding One-Time Passwords

The one-time password secret keys, code generation, and code verification are based on the industry standard HMAC-SHA1 token algorithm that is defined in the IETF RFC 6238 (http://datatracker.ietf.org/doc/rfc6238/). Each OTP is intended for use by only one user, is valid for a specific period of time, and becomes invalid after the user successfully logs in. It cannot be easily duplicated and reused elsewhere. The entered code is sent securely to CloudAccess through HTTPS (Secure HTTP) encryption on TCP port 443.

With time-based OTP, the TOTP validation server and software-token app use their respective system times to generate OTPs. The TOTP algorithm assumes that the system times are synchronized. To minimize time drift, you should configure the network time protocol (NTP) on the CloudAccess appliance so its clock stays accurate. If you cluster the CloudAccess appliances, ensure that the member nodes in the cluster point to the same centrally located time server. Users should synchronize the clocks on their mobile devices with their service providers’ networks, which are typically aligned with atomic clocks.

Time differences between the TOTP validation server and a mobile device can result in a mismatch of the OTP, and subsequent login failure. Common causes include clock time drift, network latency, and slow data entry. To allow for time differences, the Validity Time setting allows a submitted OTP to be considered valid if it matches a server-generated OTP for any time-step that occurs in a specified validity window centered on its received timestamp, plus 30 seconds. A time-step is 30-seconds.

For the TOTP tool, the default validity time setting is 5 minutes. The validity window is 2.5 minutes before and 2.5 minutes after the password's received timestamp, plus 30 seconds. You can specify integer values from 2 to 10. Shorter validity times are considered to be more secure than longer ones.

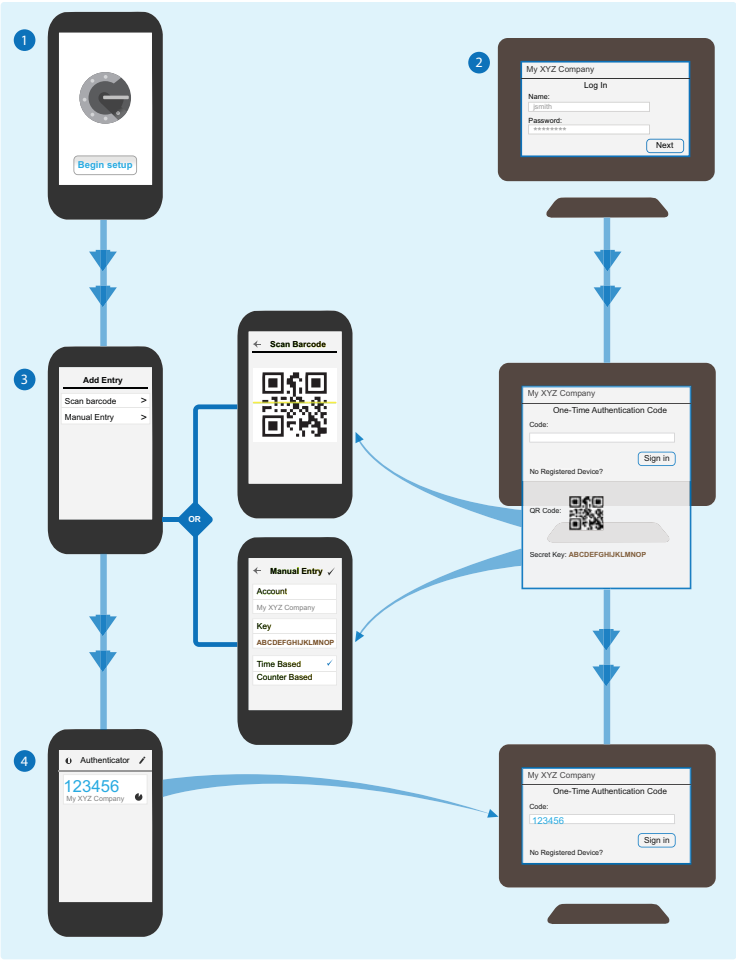
If you enable both the Google ReCAPTCHA tool and the TOTP tool in CloudAccess, ReCAPTCHA works only for the user's login password, and not for the one-time password. The ReCAPTCHA is not used if the OTP is incorrect.

5.3.2 How to Use Google Authenticator for TOTP

You can enable the TOTP validation service for one or more applications. At a user's next login, TOTP prompts the user to register a device to use for the additional authentication. If you enable all applications, the prompt occurs immediately after CloudAccess validates the user's credentials. Otherwise, the prompt occurs when the user first selects any one of the TOTP-enabled applications.

Figure 5-2 illustrates the setup for Google Authenticator on a users mobile device and the registration of the device with TOTP in CloudAccess.

Figure 5-2 *Self-Service Device Registration*



The user registers a device with CloudAccess as follows:

1. The user installs the Google Authenticator app on their mobile device.

2. On a computer, the user logs in to CloudAccess using their corporate user name and password. If the credentials are valid, CloudAccess prompts the user to register a device for one-time passwords.
3. The user registers the mobile device with their CloudAccess account.
   a. TOTP creates a secret key for the user and displays it in text and Quick Response (QR) code format in the computer web browser.
   b. On the mobile device, the user starts Google Authenticator and adds an account for CloudAccess.
   c. In Google Authenticator, the user scans the Quick Response (QR) code displayed on the browser, or manually enters the security key.
4. The Google Authenticator app generates a 6-digit code, and the user enters the code on the Authentication Code page.

The device registration process is not complete until the user submits a one-time authentication code that was generated using the secret key on the registration page, and the server validates it. If the submitted code does not validate, TOTP re-displays the registration page with the current secret key. The user can generate a new code on the mobile device, and then try again to authenticate. On successful authentication, TOTP stores the shared key with the user's identity information in the related identity source. The user's registered device and its authentication codes apply to all TOTP-enabled applications. Each session requires only a single successful authentication.

TOTP cancels the registration process if the user makes no attempt to register the device and discontinues the login by closing the tab or browser. The next time that the user logs in, CloudAccess generates a new secret key, and prompts the user to register a device with a new key.

For each log in to CloudAccess, the user runs the Google Authenticator app to generate a new 6-digit code for the CloudAccess account. When the app runs, it generates a new 6-digit authentication code every 30 seconds. This code is the one-time password that the user enters for two-factor authentication.

TOTP requires a newly generated authentication code at each subsequent login. If you enable all applications, TOTP prompts for the code immediately after CloudAccess verifies the user's credentials. Otherwise, the prompt occurs when the user first selects any one of the TOTP-enabled applications. The authentication automatically applies to other TOTP-enabled applications for that session.

An administrator can reset a device associated with a user account. This deregisters the device from the account, and invalidates the old key. The next time that the user logs in, CloudAccess generates a new secret key, and prompts the user to register a device with a new key.

5.3.3 Configuring the TOTP Tool

You can enable the Time-Based One-Time Password tool to require users to use two-factor authentication when logging in through CloudAccess.

To configure the TOTP service:

1. Log in with an appliance administrator account to the Admin page at `https://`*`appliance_dns_name`*`/appliance/index.html`.
2. Drag the **TOTP Tool** icon from the **Tools** palette to the **Tools** panel.

3 Click the **TOTP** icon on the **Tools** panel, then click **Configure**.

4 (Optional) Specify the **Validity Time**.

   Specify an integer value from 2 to 10. The default value is 5. Shorter validity times are considered more secure.

5 Click the **Applications** tab, then select the check box next to one or more applications to enable them for TOTP.

   By default, no applications are enabled for TOTP.

   When a user registers an authentication device, the device and authentication codes apply to all TOTP-enabled applications.

6 Click **OK** to save the setting and enable the TOTP tool.

7 Click **Apply** to activate the TOTP configuration.

8 Wait while the service is activated across all nodes in the cluster. Do not attempt other configuration actions until the activation completes successfully.

   In the Appliances pane, a green gear icon spins on top of each node until the activation is complete across all nodes in the cluster.

5.3.4 Registering a Mobile Device with the TOTP Tool for OTP Generation

After you enable the TOTP tool, users are prompted to register a device to use for the additional verification the next time they sign in to CloudAccess. Each user must register a mobile device for generating the user's one-time passwords. For the initial setup, the user should use a web browser on a computer other than the mobile device where the one-time passwords will be generated.

The One-Time Authentication code page displays a QR code and its equivalent secret key. The user deploys the Google Authenticator app on a mobile device, and sets up an account for CloudAccess by using the shared key. The user can scan the QR code or manually enter the key. When the app runs, it generates a new one-time password every 30 seconds.

Before registering a device, the following setup is required:

- ❒ The user must be an authorized user of CloudAccess with a valid user name and password.
- ❒ The user must have access to a computer running a supported web browser.

  For a list of supported web browsers, see Browsers in Table 2-2, "Product Requirements," on page 14.
- ❒ The user must use a supported mobile device.

  For a list of supported mobile device platforms, see Mobile Devices in Table 2-2, "Product Requirements," on page 14.
- ❒ The user must install the Google Authenticator app on the mobile device.

To register a mobile device for use with the TOTP tool:

1 (Conditional) If the Google Authenticator mobile app is not already installed on the mobile device, download and install it.

   1a Visit the app store for your mobile device.

   1b Search for Google Authenticator.

   1c Download and install the app.

2 From a computer that will not be used as the OTP device, access CloudAccess either directly or through a SAML2 redirect.

3 On the CloudAccess login page, enter your network user name and password (your normal identity source login credentials).

A message displays a QR code (and its equivalent secret key) to use for the TOTP registration.

If you are not prepared to register your mobile device at this time, you can cancel the registration process by closing the tab or your browser. On your next login, CloudAccess generates a new secret key, and prompts you to register a device with a new key.

4 On your mobile device, use the Google Authenticator app to scan the displayed QR code, and register the device with CloudAccess. You can alternatively type the secret key.

4a On your mobile device, open the Google Authenticator app.

4b Select **Settings** > **Add an account**.

4c Use either of the following methods to configure the account:

- **Scan a barcode:**
  1. Select **Scan a barcode**.
  2. Use your device's camera to scan the QR code that appears on the CloudAccess One-Time Authentication Code page.
- **Enter provided key:**
  1. Select **Time Based**.
  2. Select **Enter provided key**.
  3. Type the 16-character secret key that appears on the CloudAccess One-Time Authentication Code page. The key is case sensitive. Do not add spaces or stray characters.

4d Specify a unique name for the account.

4e Tap **Done**.

5 On the mobile device, view the 6-character code that Google Authenticator displays for CloudAccess. This is your OTP.

6 On the computer on the One-Time Authentication Code page, type the OTP, then click **Sign In**.

CloudAccess confirms that the mobile device is registered, and the login is successful.

If the code does not validate, the registration page is redisplayed with the current secret key. You can generate a new code, and try again. The code might not validate if you enter an expired code, you do not enter a code, you mistype the code, or you make an error when setting up the secret key for the account in Google Authenticator.

7 To log in to your account from the mobile device, log in to CloudAccess as described in Section 5.3.5, "Using Two-Factor Authentication at Login," on page 60.

On successful authentication, you can access the apps icons for the authorized services and resources associated with your user identity. Access is granted only for the duration of that session.

To de-register a mobile device:

1 Access CloudAccess either directly or through a SAML2 redirect.

2 Log in and authenticate as described in Section 5.3.5, "Using Two-Factor Authentication at Login," on page 60.

3 Click the **My Devices** icon.

4 In the **Registered Devices** list, select the mobile device.

5 Click the **Delete** icon for the device.

6 In the **Unregister** Device window, click OK to confirm.

At your next login, CloudAccess prompts you to register a device before you can access applications that require two-factor authentication.

5.3.5 Using Two-Factor Authentication at Login

When two-factor authentication is enabled for CloudAccess, a user must provide login credentials and a one-time authentication code to gain access to TOTP-enabled applications. The code is a 6-digit number generated for CloudAccess by the Google Authenticator app that is running on the user's mobile device. The user must have already registered the mobile device with CloudAccess, as described in Section 5.3.4, "Registering a Mobile Device with the TOTP Tool for OTP Generation," on page 58.

The user should enter the newly generated code as soon as possible after it appears in the Google Authenticator app. Each OTP is intended for use by only one user, is valid for 30 seconds, and becomes invalid after the user successfully logs in. Access is granted only for the duration of that session.

To log in to CloudAccess using two-factor authentication:

1 Access CloudAccess either directly or through a SAML2 redirect.

2 On the login page, enter your network user name and password (your normal identity source login credentials).

CloudAccess verifies the credentials against a defined identity source. If all applications require two-factor authentication, the One-Time Authentication Code page appears and prompts you to enter the code. Otherwise, CloudAccess displays the page when you first click any one of the applications that require it.

3 Use Google Authenticator to generate a new one-time password, and enter the code on the CloudAccess One-Time Authentication Code page.

If you enter the password incorrectly, you can try again with the same password until it times out. Google Authenticator generates a new OTP every 30 seconds.

On successful authentication, you can access the apps icons for the authorized services and resources associated with your user identity. Each session requires only a single successful authentication.

5.3.6 Resetting a Device (Deregistering a Device)

Each user can register a single device to use for generating one-time passwords. Resetting a device for a user's account deregisters the user's current device. The next time the user logs in, the TOTP tool creates a new secret key for the account.

An administrator can reset a device for a user account:

- To allow the user to register a different device
- To revoke access for a registered device that is lost or stolen

Information about a user's registered device and secret key is part of the user's identity information in the identity source. This information is deleted automatically if a user's identity object is permanently deleted from the identity source. The information is stored with the user's object if the user's identity object is disabled.

If the Time-Based One-Time Password tool is disabled, CloudAccess no longer prompts the users for an OTP at login. However, information about a user's registered device and secrete key continue to be stored in the users' identity objects in the identity source. The OTPs generated for the user's CloudAccess account by the Google Authenticator app are no longer needed at login.

After a device is deregistered, the OTPs generated for the user's CloudAccess account by the Google Authenticator app are no longer valid. At the user's next login, the TOTP tool generates a new secret key for the user, and the user must register a device to work with it.

To reset (deregister) a device for a user account as an administrator user:

1. Log in with an appliance administrator account to the CloudAccess Administration Console at `https://appliance_dns_name/appliance/index.html`.
2. Click the **Devices** icon.
3. In the User field, type a few characters of the user name, and then scroll in the window to select the user identity.
4. Under **One-Time Password**, click **Reset**.

5.4 Configuring the Advanced Authentication Tool for Two-Factor Authentication Using NetIQ Advanced Authentication Framework

The Advanced Authentication tool supports the use of one-time passwords (OTPs) for two-factor authentication of users as they access applications through CloudAccess. It works with the NetIQ Advanced Authentication Framework running on a member server in a domain configured as an Active Directory identity source.

With two-factor authentication, users must provide two categories of authentication factors before they can access the applications:

- **Something the user knows:** The first authentication factor requires *something the user knows*, such as the password for the user's single-sign-on user name.
- **Something the user has:** The second authentication factor requires *something the user has*, such as a device to uniquely generate or receive one-time passwords or authentication requests that can be used only for that access moment.

Two-factor authentication provides an additional layer of security that helps ensure the identity of a user and reduce the risk of unauthorized access to your applications. Users still enjoy the convenience of single sign-on, but the access is more secure.

The Advanced Authentication tool supports multiple types of authentication providers for OTP in the NetIQ Advanced Authentication Framework. You configure a separate instance of the tool for each authentication provider type you want users to use. For each authentication provider type, you can enable one or more applications, but they must be mutually exclusive of the applications that you enable in other instances. The applications must also be mutually exclusive of applications configured to use the Time-Based One-Time Password tool with Google Authenticator.

At a user's next login, the tool prompts the user for additional authentication, according to the authentication provider type enabled for the application. If you enable all applications for a single authentication provider type, the prompt occurs immediately after CloudAccess validates the user's credentials. Otherwise, the prompt occurs when the user first selects any one of the applications enabled for Advanced Authentication. The authentication automatically applies to all applications for that session that were also enabled for the same type of authentication provider.

For more information about using the NetIQ Advanced Authentication Framework and the supported authentication providers, see the NetIQ Advanced Authentication Framework (https://www.netiq.com/documentation/netiq-advanced-authentication-framework/) documentation website.

Use the information in the following sections to configure your system for Advanced Authentication:

- Section 5.4.1, "Requirements for Advanced Authentication," on page 62
- Section 5.4.2, "Configuring the Advanced Authentication Tool," on page 62

5.4.1 Requirements for Advanced Authentication

Ensure that your system meets the following requirements before you configure the Advanced Authentication tool:

- ❒ A CloudAccess 2.2 appliance, installed and configured.
- ❒ An Active Directory identity source, with the identity source connector enabled and configured.
- ❒ A server running NetIQ Advanced Authentication Framework 5.1 that is an Active Directory member server in the same domain as the Active Directory identity source.

  NOTE: CloudAccess 2.2 does not support previous versions of the NetIQ Advanced Authentication Framework.

- ❒ Before you configure the Advanced Authentication tool, ensure that you install and configure the authentication providers that you want to use on the NetIQ Advanced Authentication Framework server. For more information, see the NetIQ Advanced Authentication Framework (https://www.netiq.com/documentation/netiq-advanced-authentication-framework/) documentation website.
- ❒ The users in the domain must use the NetIQ Advanced Authentication Framework client or web user interface to enroll or re-enroll for the authenticator providers that you want them to use.
- ❒ Identify the type of authentication provider that you want to use for each of your destination applications.

5.4.2 Configuring the Advanced Authentication Tool

Before you configure the Advanced Authentication tool, ensure that your setup meets the requirements described in Section 5.4.1, "Requirements for Advanced Authentication," on page 62.

To configure the Advanced Authentication tool:

1. Log in with an appliance administrator account to the Admin page at

   `https://appliance_dns_name/appliance/index.html`

2. In the Identity Sources panel, verify that you have configured one Active Directory identity source for a domain where a member server is running NetIQ Advanced Authentication Framework.
3. Drag and drop the Advanced Authentication tool from the **Tools** palette to the **Tools** panel.

   The Configuration window opens automatically for the initial configuration. To view or reconfigure the settings later, click the Advanced Authentication tool, then click **Configure**.

4. Configure the Advanced Authentication feature:

   Authentication type: Select the type of authentication provider that you want to enable for the specified server running the NetIQ Advanced Authentication Framework.

NAAF host name/port: Specify the hostname of the Active Directory member server running NetIQ Advanced Authentication Framework. The default port number is 8232.

5 Click the **Applications** tab, then select the check box next to one or more applications that require the specification authentication provider.

   You can enable one or more applications for the specified type of authentication provider. However, you must assign each application to only one type of authentication provider.

6 Click **OK** to save the settings and enable the tool.

7 Click **Apply** to activate the configuration.

8 Wait while the service is activated across all nodes in the cluster. Do not attempt other configuration actions until the activation completes successfully.

5.5 Configuring FIDO for Two-Factor Authentication

CloudAccess supports FIDO (Fast IDentity Online) for two-factor authentication. FIDO requires that users enter their user name and password. The second factor authentication is a dongle that users must touch in order to authenticate. For more information, see the FIDO Alliance (https://fidoalliance.org/) website.

5.5.1 Requirements for FIDO

Ensure that your system meets the following requirements before you configure FIDO as an authentication method:

- ❒ A CloudAccess appliance, installed and configured.
- ❒ A FIDO supported dongle for each user.

5.5.2 Configuring the FIDO Tool

Before you configure the FIDO tool, ensure that your setup meets the requirements described in Section 5.5.1, “Requirements for FIDO,” on page 63.

To configure the FIDO tool:

1 Log in with an appliance administrator account to the Admin page at

   `https://appliance_dns_name/appliance/index.html`

2 Drag the **FIDO** tool from the **Tools** palette to the **Tools** panel.

3 Read the message that there is no configuration required.

4 Click the **Applications** tab, then select the check box next to one or more applications that require the specified authentication provider.

   You can enable one or more applications for the specified type of authentication provider. However, you must assign each application to only one type of authentication provider.

5 Click **OK** to save the settings and enable the tool.

6 Click **Apply** to activate the configuration.

7 Wait while the service is activated across all nodes in the cluster. Do not attempt other configuration actions until the activation completes successfully.

5.6 Configuring the Authentication Filter to Set Session-Based Identity Information for a User

The CloudAccess single sign-on login is designed to authenticate a user against an identity source and to share this authentication with other protected applications. The authentication process does not provide extended functions to add, remove, or manage a user's identity information for the session. To address this need, CloudAccess provides the Authentication Filter tool.

The Authentication Filter integrates with the CloudAccess single sign-on process. After the user logs in, the filter intercepts the authentication process and sends the user's identity information from the identity source to your custom authentication scripts. You can add, remove, or set values for supported identity attributes. You can also set a cookie. You can interact with the user to gather input for those changes. After all of the encoded rules and associations are complete, CloudAccess stores the modified identity information in the session cache for the web services and applications.

The Authentication Filter tool is compatible with the ExtAPI library and the ExtUI library. It works with multiple scripting languages including PHP, Java, and Perl.

For information about creating custom authentication scripts to use with the Authentication Filter, see the *Technical Reference: Authentication Filter for NetIQ CloudAccess*.

After you create your custom scripts, you must enable and configure the Authentication Filter tool in CloudAccess. The enabled filter automatically runs on each node in a CloudAccess cluster.

Before you enable the Authentication Filter, ensure that your enterprise environment meets the following requirements:

- A CloudAccess 2.1 appliance, installed and configured
- The Authentication Filter supports only applications and devices that use session-based protocols. The filter stores the altered identity attributes and values in the session attribute cache.

  The Authentication Filter does not support applications and devices that use sessionless protocols, because there is no session attribute cache to store the altered identity attributes and values. For example:

  - The OAuth protocol is a sessionless protocol. Thus, the Authentication Filter does not support applications use the OAuth Service Provider connector.
  - Mobile devices use a token-based protocol, which reestablishes the session for each transmission. Thus, there is no session attribute cache for mobile sessions, whether the connector's protocol is session-based or sessionless.
- On the ExtAPI server, create a script that uses the ExtAPI library commands to apply session-based authentication rules to an authenticated user's identity information. The Authentication Filter points to the URL for this file. The ExtAPI server is a web server that supports the programming language for the script file you create.
- If the session-based identity changes require user interaction:
  - On the ExtUI server, create a script that uses the ExtUI library commands to collect the user's session-based identity information, and return control to CloudAccess. The ExtAPI script should redirect the authentication session to the URL for this file. The ExtUI server is a web server that supports the programming language for the script file you create.
  - On the ExtAPI server, create a redirect file configured with the ExtUI script's URL.

To enable the Authentication Filter:

1. Log in with an appliance administrator account to the CloudAccess administration console at `https://appliance_dns/appliance/index.html`.
2. Drag the **Authentication Filter** icon from the **Tools** palette and drop it in the **Tools** panel.
3. In the **Tools** panel, click the **Authentication Filter** icon, then click **Configure**.
4. In the Edit External Filter window, complete the following information:

   Display name: Specify a name for the filter. This name appears on the main Admin page.

   Connects to: Specify the URL to the script that you want to run during the user SSO login.

   For example:

   `https://extapi_server_dns:port/path/extapi/index.php`

   Use HTTPS for secure SSL transfer of information. If you use an HTTP URL, information is not secure.

   Basic Auth User: (Optional) If login is required to access the URL, specify the user name to use in the basic authentication header.

   Basic Auth Password: (Conditional) If you specify a user name, specify the password for it.
5. Click **OK** to save and enable the filter settings.
6. On the Admin page, click **Apply** to activate the filter configuration.
7. Wait while the service is activated across all nodes in the cluster. Do not attempt other configuration actions until the activation completes successfully.

   In the **Appliances** panel, a green gear icon spins on top of each node until the activation is complete across all nodes in the cluster. In the **Tools** panel, a green status icon appears on the lower-left corner of the service icon. A yellow status icon appears if the URL uses HTTP instead of HTTPS because the traffic is not secure.

6 Setting Up and Managing MobileAccess

Administrators can now enable user access to SSO, proxy, and SaaS applications from supported mobile devices. For more information about supported mobile devices, see Section 2.2, “Product Requirements,” on page 14.

6.1 Introduction to MobileAccess

MobileAccess features are available for all application connectors that CloudAccess supports. Configurable options in MobileAccess include the following:

- Which applications users should be able to access.
- Whether users can access an application through a desktop browser or a mobile device, or both.
- The preferred viewer for the application on the mobile device.
- Whether users are required to provide a PIN to use the MobileAccess app on their mobile device, and if so, whether they are required to re-enter the PIN after a period of inactivity.

The MobileAccess app that end users install on their mobile devices enables them to access corporate and SaaS applications from those devices. Administrators can also make the MobileAccess app available to users in a private corporate store. Once users have installed the app and registered their device, they can access assigned applications using their corporate user name and password.

Administrators can unregister user mobile devices in the administration console. So, if a registered mobile device is lost or stolen, or an employee leaves the company, you can ensure that unauthorized users cannot access corporate resources. Users can also unregister their own mobile devices if necessary, either from their device or from the appliance administration console.

6.2 Installing and Configuring the MobileAccess Appliance

The prerequisites for the MobileAccess appliance, and the steps for installing and configuring the appliance, are the same as those for CloudAccess.

NOTE: Whether you have a full CloudAccess license or a MobileAccess-only license, you need to install only one appliance to get all features. For more information about product licensing, see Section 1.5, “Understanding Product Licensing,” on page 11.

For more information, see the following sections:

- Chapter 2, “Installing the Appliance,” on page 13
- Chapter 3, “Configuring the Appliance,” on page 29

After you have installed and configured the appliance, you can configure the MobileAccess tool. For more information, see Section 6.3, “Configuring the MobileAccess Tool on the Appliance,” on page 68.

6.3 Configuring the MobileAccess Tool on the Appliance

Once you have installed and configured the appliance, you can configure the MobileAccess tool.

To configure the MobileAccess tool:

1. Log in with an appliance administrator account to the Admin page at `https://appliance_dns_name/appliance/index.html`.
2. Drag the MobileAccess icon from the **Tools** palette to the **Tools** panel.
3. Click the MobileAccess icon on the **Tools** panel and then click **Configure**.
4. In the **Display name** field on the Mobile Application Access window, type your company name. This name appears in the bar at the top of the MobileAccess app window on users' mobile devices.
5. In the **Prompt for Password Re-authentication in x Days** field, specify how long users can continue to use an authenticated password on mobile devices before re-authentication is required.
6. (Optional) On the **PIN Required** bar, specify whether users must set a PIN for the MobileAccess app on their mobile devices, and whether they must re-enter the PIN after a period of inactivity. You can change this requirement at any time. For more information, see Section 6.7, "Understanding the MobileAccess PIN," on page 73.
7. Click **OK**.
8. Click the **Apply** button on the Admin page.
9. Wait for the apply operation to finish. (The gear stops spinning on the appliance when the operation has finished.)
10. Continue with Section 6.4, "Replacing the Default Certificate on the Appliance," on page 68.

6.4 Replacing the Default Certificate on the Appliance

You must change the default certificate that comes with the appliance before you can successfully register mobile devices. For security reasons that are well-documented, as well as for administrator and user convenience, NetIQ highly recommends that you change the default certificate on the appliance to a well-known Certificate Authority signed certificate. For more information, see Section 3.5, "Changing the Certificates on the Appliance," on page 33.

Before you change the certificate on the appliance, ensure that your environment meets the following requirements:

- ❒ The appliance must be installed and running with a DNS entry that points to it.
- ❒ The certificate must be at least 2k key size (4k preferably) using SHA256.
- ❒ The certificate must be signed by a Certificate Authority, preferably a well-known Certificate Authority. If you choose to use a self-signed certificate, it must be flagged as a certificate authority.

If you use a self-signed or non-public CA-signed certificate, users must also install the certificate on their mobile devices. For more information, see the following topics:

- Section 6.4.1, "Generating a Self-Signed Certificate," on page 69
- Section 6.4.2, "Installing a Self-Signed Certificate on the Mobile Device," on page 69

6.4.1 Generating a Self-Signed Certificate

You can generate a self-signed certificate and use it on the appliance, but if you do so, you must also perform the steps in Section 6.4.2, “Installing a Self-Signed Certificate on the Mobile Device,” on page 69 to ensure that you can successfully register mobile devices. You can run the Java 7 keytool on a computer other than the appliance to generate the certificate.

To generate a self-signed certificate:

1. Using the Java 7 keytool, use the following commands replacing *name* and *appliance_dns_name*:

   ```
   keytool -genkeypair -keystore name.p12 -storepass changeit -sigalg
   SHA256withRSA -keyalg RSA -keysize 4096 -dname "CN=appliance_dns_name" -
   validity 365 -storetype pkcs12 -ext bc=ca:true
   ```

   name can be anything you want, as long as it is the same between the two commands, and you can find it when you want to upload it.

   appliance_dns_name must be the DNS name of the appliance.

   The output of this command is a `.p12` format file. You can use this file to replace the default certificate on the appliance. (Use the password of `changeit` when the administration console prompts for it.) For more information, see Section 3.5, “Changing the Certificates on the Appliance,” on page 33.

2. To get the public certificate from that keyfile (which you will use when you perform the procedure in Section 6.4.2, “Installing a Self-Signed Certificate on the Mobile Device,” on page 69) use the following command, replacing *name* with the same value from above:

   ```
   keytool -export -keystore name.p12 -storetype pkcs12 -alias mykey -file
   name.cer -storepass changeit
   ```

   The output of this command is a `name.cer` file that you can use later.

6.4.2 Installing a Self-Signed Certificate on the Mobile Device

This procedure is required only if you used the commands in Section 6.4, “Replacing the Default Certificate on the Appliance,” on page 68 to generate the certificate. If you are using a certificate signed by a well-known Certificate Authority, you can skip this section.

To install a self-signed certificate on the mobile device:

1. Take the `name.cer` file that you generated in Section 6.4, “Replacing the Default Certificate on the Appliance,” on page 68 and email it to the user who has an email account configured on the mobile device. Alternatively, you could put it on a web or FTP site that is accessible from the mobile device.
2. Open the email (or web/FTP site) on the mobile device and tap the certificate attachment.
3. In the Install Profile window, tap **Install**.
4. Read the warning and then tap **Install**.
5. Verify that the certificate says “Trusted” with a green check mark in the Profile Installed window.
6. (Conditional) If the certificate is not trusted, something is wrong with the certificate and the MobileAccess application will not work. Go back and try to generate the certificate again.
7. Tap **Done**.
8. Verify that this procedure worked by entering the appliance DNS name in the Safari address bar and ensuring that there is no warning about an untrusted certificate.

NOTE: This step does not currently work in Chrome.

This certificate is installed in the Settings > General > Profiles page on the mobile device and can be removed from that location on the device.

The server certificate and the trusted root certificate need to be at least 2k in size.

Once you have replaced the default certificate on the appliance, you can continue with MobileAccess installation and configuration. For more information, see Section 6.5, “Installing MobileAccess on a Mobile Device,” on page 70.

6.5 Installing MobileAccess on a Mobile Device

Once an administrator has enabled and configured the MobileAccess tool in the administration console, users must install the MobileAccess app on supported mobile devices before they can access SaaS, SSO, or proxy applications that have been configured for mobile access. If users do not yet have the MobileAccess app installed, they are prompted to do so and redirected to the App Store. Access to the administration console from mobile devices is not supported.

To install the MobileAccess app on a mobile device:

1. Access the App Store on the mobile device.
2. Search for the NetIQ MobileAccess app.
3. Tap **Install**.
4. Continue with Section 6.6, “Registering a Mobile Device with the Appliance,” on page 70.

6.6 Registering a Mobile Device with the Appliance

Users can install the NetIQ MobileAccess app on a mobile device to access the single sign-on services of MobileAccess providers and CloudAccess providers. A user can register a device with multiple providers by setting up separate accounts for each one. If a user registers a device with multiple providers, the user must select the account to use for a session from the **Accounts > Providers** list. By default, the app connects the user to the first provider in the list.

Before users can register their mobile device with the appliance, the following prerequisites must be met:

- A MobileAccess appliance or a CloudAccess appliance, installed and configured.
- The MobileAccess tool, enabled and configured on the appliance.
- A signing certificate configured on the appliance.
- Applications, configured and entitlements are configured by mapping authorizations for applications to roles (groups) in the identity source.
- The MobileAccess app, installed by users on their mobile devices. During registration, the mobile device should be connected to a network that does not use HTTP proxy.

6.6.1 iOS Devices

To register a mobile device using a link from the appliance administrator:

1. The administrator sends users an email with a link for the CloudAccess appliance. The link looks like this:

`comnetiqauth://x-callback-url/register?providerUrl=https://appliance_dns_name/`

2. The user opens the email from the mobile device.
3. The user taps the link.

   The link launches the application with the **Provider** value filled in.
4. The user taps **SignIn** to begin the registration process.
5. On the CloudAccess login page, the user types their corporate credentials for the account, and taps **Sign in**.
6. (Conditional) If the administrator has specified that a PIN is required for the MobileAccess app, the user creates a PIN to be entered when the app launches.

   The user's mobile device is registered.

To register a mobile device manually:

1. The administrator provides users the login URL for the CloudAccess appliance. For example:

   `https://appliance_dns_name`
2. The user launches the MobileAccess app on their mobile device.
3. The user taps **Accounts**, then taps **Plus** (**+**) to add a new account.
4. The user types the **Provider URL** for the account, then taps **SignIn** to begin the registration process.
5. On the CloudAccess login page, the user types their corporate credentials for the account, and taps **Sign in**.
6. (Conditional) If the administrator has specified that a PIN is required for the MobileAccess app, the user creates a PIN to be entered when the app launches.

   The user's mobile device is registered.

To verify that the device is registered with the appliance:

1. The user taps **Accounts** at the bottom of the app window.
2. The user can verify that the device is listed under **Devices**, along with the time of registration.

   The device name is the name that is set on the **Settings > General > About** window of the mobile device. Users can change this name, but it is not configurable by the administrator of the CloudAccess appliance.

Tapping the **All Apps** icon displays the icons for configured applications. This page is empty until the administrator configures application connectors to be accessible from mobile devices.

Tapping the **Favorites** icon displays the icons for apps that the user chooses as their favorites. This page is empty until the user adds apps from the All Apps page to the Favorites page.

To add an app to Favorites:

1. On the All Apps page, long-press any app until all of the apps begin to shake.
2. Select the **X** on an app to add it to **Favorites**.
3. Tap **Home** to end the selection process.

After the user adds apps to **Favorites**, MobileAccess opens to the Favorites page. The user can view all applications by tapping **All Apps**.

To deregister a mobile device from the appliance:

1. The user taps **Accounts** at the bottom of the app window.

2 Under **Devices**, the user swipes left on the device name to expose the **Delete** option.

3 The user taps **Delete**.

The user's mobile device is deregistered.

6.6.2 Android Devices

To register a mobile device using the app UI:

1 The administrator provides users the login URL for the CloudAccess appliance. For example:

`https://appliance_dns_name`

2 The user launches the MobileAccess app on their mobile device.

3 If this is the first time the user opens the app, it presents the MobileAccess License Agreement. The user reads the agreement and taps **Accept** to continue.

4 The user taps the **Tools** icon, then taps **Accounts > Add Accounts**.

5 The user types the DNS name or URL for the account, then taps **Sign in** to begin the registration process.

6 On the CloudAccess login page, the user types their corporate credentials for the account, and taps **Sign in**.

7 (Conditional) If the administrator has specified that a PIN is required for the MobileAccess app, the user creates a PIN to be entered when the app launches.

The user's mobile device is registered.

To register a mobile device using the device settings:

1 The administrator provides users the login URL for the CloudAccess appliance. For example:

`https://appliance_dns_name`

2 The user launches the **Settings** tool on their mobile device.

3 The user taps **Add Account**.

4 In the list of installed apps, the user taps **MobileAccess**.

5 If this is the first time the user opens the app, it presents the MobileAccess License Agreement. The user reads the agreement and taps **Accept** to continue.

6 The user types the DNS name or URL for the account, then taps **Register** to begin the registration process.

7 On the CloudAccess login page, the user types their corporate credentials for the account, and taps **Sign in**.

8 (Conditional) If the administrator has specified that a PIN is required for the MobileAccess app, the user creates a PIN to be entered when the app launches.

The user's mobile device is registered. The device opens to the landing page for the provider. This page is empty until the administrator configures application connectors to be accessible from mobile devices.

To verify that the device is registered with the appliance:

1 The user taps the Tools icon to view the Settings page.

2 The user taps **Manage Devices**.

3 The user can verify that the device is listed under **Devices**, along with the time of registration.

   The device name is the name that is set on the **Settings > General > About** window of the mobile device. Users can change this name, but it is not configurable by the administrator of the CloudAccess appliance.

To deregister a mobile device from the appliance using the app interface:

1 The user taps the **Tools** icon to view the Settings page.
2 Under **Accounts**, the user taps **Remove account:** *appliance_dns_name*.

   The user's mobile device is deregistered.

To deregister a mobile device from the appliance using device settings:

1 The user launches the **Settings** tool on their mobile device.
2 Under **Accounts**, the user taps **MobileAccess**.
3 The user taps the menu, then taps **Remove Account**.

   The user's mobile device is deregistered.

6.7 Understanding the MobileAccess PIN

MobileAccess administrators can require users to set a PIN on their mobile devices as a security measure to prevent unauthorized users from accessing protected resources through the MobileAccess app. Administrators can also specify whether users must re-enter the PIN after a period of inactivity on the device.

Users must install the MobileAccess app on the mobile device before they can set the PIN. If a PIN is required, the MobileAccess app prompts users to set the PIN the first time they open the app. Otherwise, users can set, change, or remove the PIN any time by accessing the Settings page from the MobileAccess app.

NOTE: The MobileAccess PIN is unrelated to the built-in device passcode, which is designed to protect other resources on the mobile device.

Even if the administrator does not require users to set a PIN, users can optionally set a PIN on their device. The PIN can be different for each mobile device the user registers. The PIN is not stored anywhere other than on the device itself.

Administrators can change the **PIN Required** setting any time in the administration console. If the administrator specifies that a PIN is required after a mobile device has already been registered, the next time the user launches the MobileAccess app on the mobile device, MobileAccess prompts the user to set a PIN. The app then prompts the user for that PIN each subsequent time the user accesses the app. If the administrator initially requires users to set a PIN and then changes that requirement, users can remove the PIN from their device. However, MobileAccess does not notify users if a PIN is no longer required.

Whether the MobileAccess administrator requires users to set a PIN or a user chooses to set a PIN, by default users can enter their PIN incorrectly five times. On the fifth attempt, the application unregisters the mobile device and removes the current PIN. The user must then reregister the device and reset the PIN. For more information, see Section 6.6, "Registering a Mobile Device with the Appliance," on page 70.

6.8 Managing Mobile Devices

Administrators who have the Device Administrator role can manage and unregister user devices in the appliance administration console. So, if a registered mobile device is lost or stolen, or an employee leaves the company, you can ensure that unauthorized users cannot access corporate resources.

Users can also unregister their own mobile devices, either from their device or from the administration console. A mobile device that has previously been unregistered can be reregistered by the same user. However, for a different user to use the unregistered mobile device, the user must delete and reinstall the MobileAccess app on the device before reregistering the device.

Use the information in the following sections to help you manage mobile devices:

- Section 6.8.1, “Unregistering Mobile Devices from the Administration Console,” on page 74
- Section 6.8.2, “Unregistering a Mobile Device from the Device,” on page 74
- Section 6.8.3, “Deleting and Reinstalling the MobileAccess App on a Device,” on page 75

6.8.1 Unregistering Mobile Devices from the Administration Console

The Devices page lists the devices for the logged-in user by default. If you are logged in with an account that has the Device Administrator role assigned, you have the option to search for and unregister devices that are registered to other users. If you log in with a regular user account, you can view and manage only your own registered devices.

To unregister mobile devices from the administration console as a Device Administrator user:

1. Log in to the Admin page of the console at `https://appliance_dns_name/appliance/index.html`.
2. Click **Devices** at the top of the page.
3. (Conditional) If you want to search for the devices belonging to a particular user, enter the user name in the **User** field.
4. Click the trash can icon next to the device you want to unregister, then click **OK** on the confirmation message.

To unregister mobile devices from the Devices page as a regular user:

1. Browse to `https://appliance_dns_name/appliance/Devices.html`.
2. Enter your login credentials when prompted.
3. Click the trash can icon next to the device you want to unregister, then click **OK** on the confirmation message.

Once a mobile device has been unregistered, the device can be registered to a new user. However, the MobileAccess app on the device must first be deleted and reinstalled. For more information, see Section 6.8.3, “Deleting and Reinstalling the MobileAccess App on a Device,” on page 75.

6.8.2 Unregistering a Mobile Device from the Device

Users who have previously registered a mobile device can unregister the device if necessary.

To unregister a mobile device from the device:

1. The user launches the MobileAccess app on the device.
2. (Conditional) If a PIN has been set up, the user enters the correct PIN when prompted.
3. The user taps the **Settings** option at the bottom of the application.
4. The user taps the **Unregister** button.

   The device is now unregistered.
5. (Conditional) If the device is going to be reregistered to a different user, the user should clear browser cookies on the device before the device is reregistered.

NOTE: Users can uninstall the MobileAccess app on a mobile device once the device has been unregistered. However, if the MobileAccess app is uninstalled without the device first being unregistered, the device continues to appear on the Devices page of the administration console. The administrator or user can delete the device from the Devices page.

6.8.3 Deleting and Reinstalling the MobileAccess App on a Device

Once a mobile device has been unregistered, the MobileAccess app on the device must be deleted and reinstalled before a different user can reregister the device.

1. Follow Apple's instructions to uninstall the MobileAccess app at the following web page:

   (http://www.apple.com/support/iphone/assistant/application/#section_5)
2. Reinstall the MobileAccess app. For more information, see Section 6.5, "Installing MobileAccess on a Mobile Device," on page 70.

7 Configuring Connectors

CloudAccess provides multiple connectors to SaaS applications. The connector for Google Apps for Business, the connector for Salesforce, and the connector for Microsoft Office 365 enable both account provisioning and single sign-on (SSO). The other available connectors, which are downloadable from NetIQ Downloads (https://dl.netiq.com/), provide only single sign-on capability.

The connectors for Google Apps and Salesforce are embedded in the appliance and are visible on the Admin page of the administration console as soon as you have initialized the appliance. The connector for Office 365 is included with the CloudAccess appliance. However, the administration console displays the connector only after you have installed the connector on the Windows server.

IMPORTANT: The connectors for Google Apps, Salesforce, and Office 365 are CloudAccess-only features and are not included in the MobileAccess-only license. For more information about product licensing, see Section 1.5, “Understanding Product Licensing,” on page 11.

CloudAccess also ships with the connector for NetIQ Access Manager, the connector for Bookmarks, the connector for OAuth2 Resources, and the connector for Simple Proxy.

For more information, see the *NetIQ® CloudAccess Connectors Guide*.

7.1 Overview of CloudAccess Connectors

CloudAccess uses connectors to provide single sign-on (SSO) access for users to web resources through CloudAccess. CloudAccess authenticates the users against your identity sources. When the user accesses the link for an application through CloudAccess, CloudAccess shares the authenticated user’s identity information with the destination application to establish the user’s session. Each user can access only the links they are authorized to use, according to the entitlements you set for each application.

- Section 7.1.1, “Understanding Single Sign-On Methods,” on page 78
- Section 7.1.2, “Connectors for Federated Single Sign-On and Provisioning,” on page 82
- Section 7.1.3, “Connectors for Federated Single Sign-On,” on page 83
- Section 7.1.4, “Connectors for Basic Single Sign-On,” on page 83
- Section 7.1.5, “Connector for OAuth 2.0 Single Sign-On,” on page 83
- Section 7.1.6, “Connector for Simple Proxy Single Sign-On,” on page 84
- Section 7.1.7, “Connector for Bookmarks,” on page 84
- Section 7.1.8, “Application Catalog,” on page 84
- Section 7.1.9, “Custom Connectors,” on page 84
- Section 7.1.10, “License Information for Connectors,” on page 85

7.1.1 Understanding Single Sign-On Methods

CloudAccess supports single-sign for a variety of web services and applications that have different authentication requirements. The method used for single sign-on depends on the security requirements and capabilities of each destination resource.

- "Federated Single Sign-On with SAML 2.0 or WS-Federation" on page 78
- "Basic Single Sign-On" on page 79
- "OAuth 2.0 Single Sign-On" on page 81
- "Simple Proxy Single Sign-On" on page 81
- "Bookmarks" on page 82

Federated Single Sign-On with SAML 2.0 or WS-Federation

Federated single sign-on relies on a trust relationship between an identity provider and a service provider to give a user access to a protected web service or application through CloudAccess. Open standards for federation include SAML 2.0 (Security Assertion Markup Language), WS-Federation (Web Services Federation), and SAML 2.0 Inbound. They provide a vender-neutral means of exchanging user identity, authentication, and attribute information. The service provider trusts the identity provider to validate the user's authentication credentials, and to send identity information about the authenticated user. The service provider accepts the data and uses it to give the user access to the destination service or application. This data exchange is transparent for the user. It allows the user to access the web service or application without providing an additional password.

The following describes the SSO experience for trusted access to an application through CloudAccess:

1. The user provides login credentials directly to CloudAccess, such as their corporate user name and password.
2. CloudAccess authenticates the user's credentials against the identity sources.
3. CloudAccess presents the landing page to the user with links to applications that the user is entitled to use.
4. When a user clicks an application's link, as the identity provider, CloudAccess produces an authentication assertion or token for the service provider that contains the identity attributes needed for the user request.
5. The service provider consumes the assertion or token to establish a security context for the user.
6. The service provider validates the assertion and authorizes the resource request.
7. The service provider establishes a session with the user.

CloudAccess can also provide authentication when the user initiates access to the application from the service provider.

The following describes the SSO experience for trusted access to an application initiated from the service provider:

1. The user attempts to log in to application.
2. The login is redirected to CloudAccess.
3. CloudAccess prompts the user for the user name and password. Or, if Kerberos is configured, CloudAccess performs seamless authentication.
4. CloudAccess verifies the user name and password using the identity sources. Or, if Kerberos is configured, CloudAccess validates the Kerberos token.

5. CloudAccess provides an assertion to the application service provider.
6. The service provider validates the assertion and allows the user to access the application.

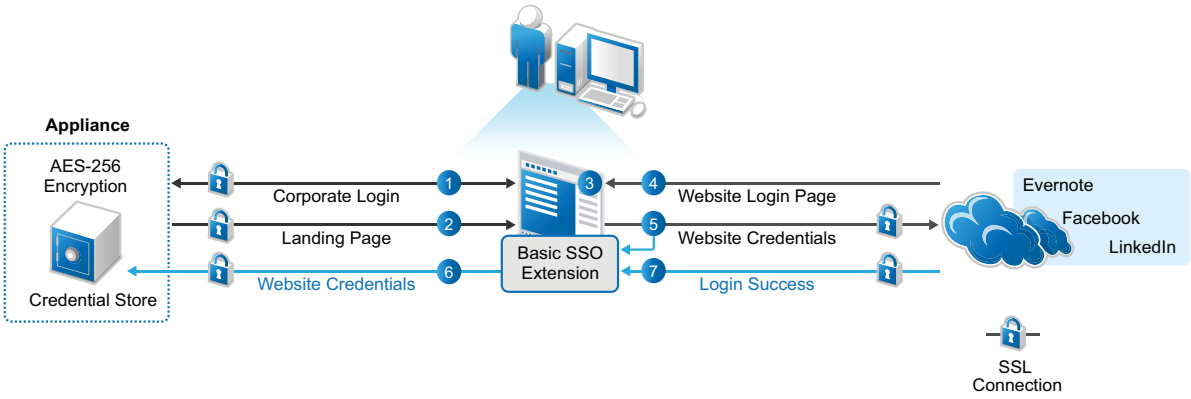
Basic Single Sign-On

Basic single sign-on provides an internal credentials store where users can save their credentials for third-party websites that require a password be sent at login. The destination website's login page must use HTML Forms as the main point of interaction with the user. A user typically has a site-specific user name and password for each destination website. CloudAccess stores the user's credentials for each site in AES-256 encrypted format. After a user authenticates to CloudAccess, the user can access a website without manually re-entering the user's credentials for the site.

Basic SSO connectors work with the Basic SSO extension running on the user's computer to securely collect, store, retrieve, and replay the user credentials for a destination website. Users must log in to the website once in order for the extension to capture and store the credentials in the CloudAccess credential store. The user can choose whether to store the credentials for each destination website. If the user does not allow credentials to be saved for a website, the user must enter the site's credentials for each session.

Figure 7-1 depicts the user experience when the user clicks the appmark for a Basic SSO application.

Figure 7-1 *User's First-Time Login to the Website with Basic SSO*



The following describes the experience for Basic SSO the first time the user accesses the app:

1. In a supported browser, the user logs in to the CloudAccess login page using their corporate credentials.
2. The user sees the available applications on the landing page.
3. The user clicks the appropriate application icon.

   If the Basic SSO extension is not installed on the computer:

   a. The connector prompts the user to install the Basic SSO extension.
   b. The user accepts the prompt, and follows the onscreen instructions to install the extension.
   c. The user returns to the landing page and clicks the appropriate application icon again.
4. A new tab opens for the login page of the application.
5. The user enters their user name and password for the destination website.

   The user must enter this separate user name and password once.

6. The extension asks if the user wants the credentials to be saved by CloudAccess, and the user allows the credentials to be saved.
   a. The extension captures the user name and password, and sends them to CloudAccess over an SSL connection.

      The extension obfuscates the user name and password with Base64 encoding before transmission.
   b. CloudAccess encrypts the site-specific credentials with AES-256 encryption, and then stores the encrypted data in the credential store that is part of the appliance.

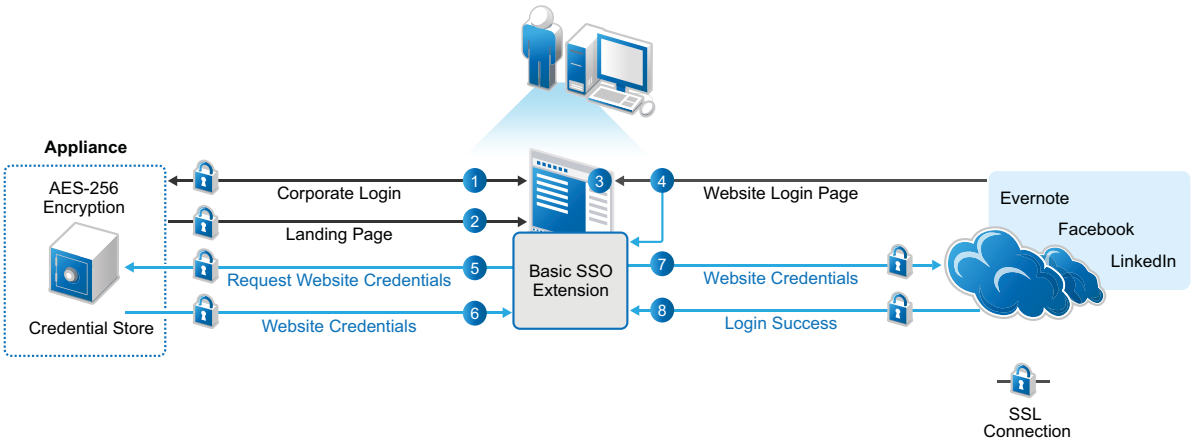
      The appliance encrypts the user name and password with an encryption key that is unique per user.
7. The website returns a success or failure indicator for the login.

   If the login succeeds, the browser opens to the application's website over an SSL connection.

   If the login fails, the browser returns the user to the website's login page to try again.

After the user allows the password to be stored securely, the user experiences single-sign-on access to the application in subsequent sessions. Figure 7-2 depicts the user experience when the user clicks the appmark for a Basic SSO application and the user's credentials are available in the credentials store.

Figure 7-2 *User's Single Sign-On Access to a Website with Basic SSO*



The following describes the experience for Basic SSO after the user stores credentials:

1. The user logs in to the CloudAccess login page using their corporate credentials.
2. The user sees the available applications on the landing page.
3. The user clicks the appropriate application icon.
4. A new tab opens for the login page of the application.
5. The Basic SSO extension requests that CloudAccess retrieve the user's user name and password for the site from the credential store.
6. CloudAccess retrieves the site-specific encrypted credentials from the credential store, decrypts them, and then sends the user name and password over an SSL connection, and replays their entry in the browser.

   CloudAccess obfuscates the user name and password with Base64 encoding before transmission.
7. If the replayed username and password are valid, the browser opens to the application's website over an SSL connection. To the user, the retrieval and replay gives a single sign-on experience.

If the user's login credentials are invalid, the browser displays the login page with the failed login error, prompting the user to log in again. The new credentials are stored using the same process as for the initial setup.

8. The website returns a success indicator for the login, and the browser opens to the application's website over an SSL connection.

OAuth 2.0 Single Sign-On

OAuth 2.0 single sign-on provides simple authenticated access to a protected web service through CloudAccess. CloudAccess behaves as an OAuth 2.0 Authorization Server and Resource Server to provide user authentication and all OAuth2 token creation and validation for access. It uses the Authorization Code flow as detailed in the *OAuth 2.0 Authorization Framework (IETF RFC 6749)* (http://tools.ietf.org/html/rfc6749#section-4.1) document.

CloudAccess supports OAuth 2.0 access in service-provider mode. End users can access the protected resource by browsing to the URL of the OAuth client application. For example, the user can enter the URL directly into the browser and be redirected to log in to CloudAccess, or they can use a bookmark or the landing page appmark after logging in to CloudAccess.

The following describes the experience for OAuth 2.0 access to an application by browsing to the URL:

1. The user accesses the protected resource by entering the URL directly in the browser.
2. The user is redirected to the CloudAccess login page.
3. The user provides login credentials to CloudAccess, such as their corporate user name and password.
4. CloudAccess authenticates the user's credentials against the identity sources.
5. CloudAccess validates the OAuth2 token for the client.
6. The user gains access to the resource.

The following describes the experience for OAuth 2.0 access to an application through CloudAccess:

1. The user provides login credentials directly to CloudAccess, such as their corporate user name and password.
2. CloudAccess authenticates the user's credentials against the identity sources.
3. CloudAccess presents the landing page to the user with links to applications that the user is entitled to use.
4. The user clicks the bookmark or the landing page appmark for the application.
5. CloudAccess validates the OAuth2 token for the client.
6. The user gains access to the resource.

Simple Proxy Single Sign-On

Simple proxy single sign-on provides reverse proxy access to your enterprise web service through CloudAccess. If the web service requires user identity information to control access or content, you can configure the connector to inject the authenticated user's identity attributes in query strings and HTTP headers sent to the web service. However, the connector cannot be used to provide single sign-on for web services that require passwords for access. This proxy solution cannot inject the password. It does not support site redirects.

The following describes the experience for simple proxy access to a web service through CloudAccess:

1. The user provides login credentials directly to CloudAccess, such as their corporate user name and password.
2. CloudAccess authenticates the user's credentials against the identity sources.
3. CloudAccess presents the landing page to the user with links to applications that the user is entitled to use.
4. The user clicks the appmark for the application.
5. (Conditional) CloudAccess sends identity information about the user in query strings and headers.
6. The website validates the resource request.
7. The user gains access to the resource.

Bookmarks

In CloudAccess, you can create bookmarks to web applications through CloudAccess that do not require additional passwords. The bookmarks are accessible from the browser landing page or directly from the MobileAccess app on users' mobile devices.

The following describes the experience for bookmark access to a web application through CloudAccess:

1. The user provides login credentials directly to CloudAccess, such as their corporate user name and password.
2. CloudAccess authenticates the user's credentials against the identity sources.
3. CloudAccess presents the landing page to the user with links to applications that the user is entitled to use.
4. The user clicks the appmark for the bookmark.
5. The user gains access to the resource.

7.1.2 Connectors for Federated Single Sign-On and Provisioning

CloudAccess provides three connectors that enable federated single sign-on and logout as well as account provisioning. The connectors ship with the appliance. You can use these connectors if you have a CloudAccess license as well as an account with the destination service.

- "Connector for Google Apps for Business"
- "Connector for Microsoft Office 365 (SAML 2.0 or WS-Federation)"
- "Connector for Salesforce (SAML 2.0)"

After you initialize the appliance, the connectors for Google Apps and Salesforce are automatically visible in the Applications palette on the Admin page of the administration console. However, the connector for Office 365 is not visible in the palette until you install the connector on the Windows Management Server.

Provisioning is available for users in your corporate identity sources for Active Directory, eDirectory, and JDBC. You must map authorizations for the appropriate roles (groups) to enable their entitlements to the applications. Users must log in with a corporate identity in order to access their provisioned account.

7.1.3 Connectors for Federated Single Sign-On

CloudAccess provides additional connectors that you can use for federated single sign-on to web services and applications through CloudAccess. The connectors support either the SAML 2.0 protocol or the WS-Federation protocol.

The connector for NetIQ Access Manager ships with the appliance. You can use this connector if you have a CloudAccess license or a MobileAccess-only license as well as an account for Access Manager. For more information, see "Connector for NetIQ Access Manager".

You can download additional connectors from NetIQ Downloads (https://dl.netiq.com/). For more information, see the *NetIQ CloudAccess Connectors Guide*.

You can also create custom connectors for federated single sign-on and logout by using the NetIQ Access Connector Toolkit. For more information, see "Creating Custom Connectors".

After you download a connector or create a custom connector, you must import it to CloudAccess to make it available in the **Applications** palette in the CloudAccess administration console. You can use these connectors if you have a CloudAccess license as well as an account with the destination service.

7.1.4 Connectors for Basic Single Sign-On

CloudAccess provides many connectors for Basic Single Sign-on (SSO). They allow users to access web services that use forms-based authentication and require that the user's password be sent at login. Examples include social media sites such as Evernote, LinkedIn, and Facebook. Basic SSO connectors work with the Basic SSO extension for supported browsers running on the user's computer.

CloudAccess supports using multiple connectors for Basic SSO. Each instance points to a different destination website. You can use these connectors if you have a CloudAccess license. Users have individual accounts with the destination services.

You can import the connectors for Basic SSO from the Application Catalog.

You can also create custom connectors for Basic SSO by using the NetIQ Access Connector Toolkit. For more information, see "Creating Custom Connectors". After you create a custom connector, you must import it to CloudAccess to make it available in the **Applications** palette in the CloudAccess administration console. For more information, see "Connectors for Basic SSO".

7.1.5 Connector for OAuth 2.0 Single Sign-On

CloudAccess provides a connector for OAuth2 Resources that allows single sign-on with simple OAuth 2.0 authenticated access to a protected web service through CloudAccess. The connector ships with the appliance.

CloudAccess supports using multiple instances of the connector for OAuth2 Resources. Each instance points to a different destination OAuth 2.0 resource, or to a set of OAuth 2.0 resources that have the same authentication requirements. You can use this connector if you have a CloudAccess license as well as an account with the destination service.

For more information, see "Connector for OAuth2 Resources".

7.1.6 Connector for Simple Proxy Single Sign-On

CloudAccess provides a connector for Simple Proxy that gives users reverse proxy access to your enterprise web service through CloudAccess. The connector ships with the appliance.

CloudAccess supports using multiple instances of the connector for Simple Proxy. Each instance points to a different destination website path. You can use this connector if you have a CloudAccess license or a MobileAccess-only license as well as access to the destination web path.

For more information, see "Connector for Simple Proxy".

7.1.7 Connector for Bookmarks

The connector for Bookmarks is a container for simple bookmarks to applications that do not require additional passwords for access. The connector ships with the appliance. You can use this connector if you have a CloudAccess license or a MobileAccess-only license as well as access to the destination web service.

For more information, see "Connector for Bookmarks".

7.1.8 Application Catalog

NetIQ provides connectors for Basic SSO in the NetIQ Application Catalog. You can browse or search the catalog for appropriate connectors to import to your appliance. You can access the catalog from the Applications palette in the administration console.

7.1.9 Custom Connectors

CloudAccess provides the NetIQ Access Connector Toolkit (ACT) that allows you to create custom connectors. If you need help creating a custom connector to use with CloudAccess, Priority Support customers have the option to open a service request with NetIQ Technical Support (NTS) (http://www.netiq.com/support). NTS is available to provide toolkit support as well as to configure the connectors to work with integrated applications. Additional information from the SaaS provider is usually required.

NOTE: Before you contact NetIQ Technical Support, please complete the appropriate worksheet for the connector type that you want to create. See "Custom Connector Worksheets".

The Access Connector Toolkit facilitates custom connector development efforts without coding or scripting. You can create connectors for identity-aware SaaS applications that support federated single sign-on and logout or that support basic single sign-on. You can use the toolkit and custom connectors if you have a CloudAccess license as well as appropriate accounts with the destination services.

For more information, see "Creating Custom Connectors".

7.1.10 License Information for Connectors

A CloudAccess license entitles you to use any of the connectors described in this guide, including custom connectors.

A MobileAccess-only license entitles you to use only the following three connectors on the **Applications** palette in the CloudAccess administration console. All other connectors, including custom connectors, are CloudAccess-only features and require a CloudAccess license.

- "Connector for NetIQ Access Manager"
- "Connector for Bookmarks"
- "Connector for Simple Proxy"

For more information, see "Understanding Product Licensing".

7.2 Viewing Connectors for Applications

CloudAccess displays the connectors on the Admin page of the administration console:

- **Applications palette:** Displays unconfigured connectors that ship with the appliance or that you have downloaded or imported.
- **Applications panel:** Displays configured connectors for the web services or applications that you want to make available to users.

7.3 Providing Access to Applications for Users

After you configure the connectors for applications and set entitlements for users, you must provide a way for users to access the applications. CloudAccess provides the following portal pages for the users:

- **Login page:** The login page allows users to enter their corporate credentials for authentication, such as user name and password. CloudAccess authenticates a user against your identity sources.

  The login page also exposes the authentication extensions of the appliance, such as Google reCAPTCHA.

  The URI for the login page is the public DNS name of the appliance. Provide this URl to your users.

  `https://appliance_dns_name`

- **Authentication code page:** The authentication code page requires users to enter a one-time password if a one-time password extension is enabled for one or more applications.

  If you enable the same extension for all applications, users see the authentication code page immediately after logging in. Otherwise, they see it when they click any one of the enabled applications.

  For example, you can enable an extension for the Time-Based One-Time Password with Google Authenticator or the One-Time Password with the OATH OTP or Smartphone option for NetIQ Advanced Authentication.

- **Landing page:** The landing page contains the appmarks (linked icons) for accessing the applications that a user is entitled to use. The landing page appears after a user enters valid credentials and responds successfully to any additional authentication prompts.

This page displays appmarks for an application only after the related connector is configured properly and if the user has an entitlement for the application. When an authenticated user clicks an appmark, CloudAccess shares identity information about the user with the application in order to establish the user's session.

In a desktop browser, users can personalize the Card Style used to display appmarks. Right-click the appmark to view or copy the Login URL. Mouse over the appmark to view the application name.

For more information, see Section 7.5, "Configuring Appmarks for Connectors," on page 91.

7.4 How CloudAccess Provisions User Accounts

The connector for Google Apps for Business, the connector for Salesforce, and the connector for Microsoft Office 365 support both account provisioning and single sign-on. These three connectors are delivered with the appliance. You can use these connectors if you have a CloudAccess license as well as an account with the destination service.

- Section 7.4.1, "Requirements for Provisioning," on page 86
- Section 7.4.2, "Understanding Provisioning," on page 86
- Section 7.4.3, "Samples of Account Creations," on page 87
- Section 7.4.4, "CloudAccess Naming Convention for Newly Provisioned Accounts," on page 88
- Section 7.4.5, "Matching Criteria for Merging Existing Accounts," on page 89

7.4.1 Requirements for Provisioning

The connectors that support SSO and provisioning can provision accounts for users in your corporate identity sources for Active Directory, eDirectory, and JDBC. You must map authorizations for the appropriate roles (groups) to enable their entitlements to the applications. Users must log in with a corporate identity in order to access their provisioned account.

The connectors cannot provision accounts for users in a Self-Service User Store (SSUS) identity source or a SAML 2.0 Inbound (SAML2 In) internal identity store.

IMPORTANT: Although CloudAccess does not prevent it, do not create policy mappings between provisioning applications and SSUS or SAML2 In identity sources.

7.4.2 Understanding Provisioning

CloudAccess creates a new account or merges an existing account in the SaaS applications for users who are members of groups in the identity sources that you map for entitlements to the applications. This is called *provisioning*.

Account provisioning occurs in two ways:

- **Automatic:** CloudAccess provisions an account for users who are members of groups that are entitled to use the application.
  - If approval is not required when you configure policy mapping, an account is provisioned when you map authorizations for the SaaS applications to the identity source roles.
  - If approval is required, the account is not provisioned until the request is approved.

- **User-controlled:** You can configure the SaaS connectors to allow users control of when their accounts are provisioned. A configuration option is available on the connector for Google Apps and the connector for Salesforce to **Prompt users for an existing account before provisioning**.

  When you select this option, users have two choices during their initial attempt to access the SaaS application using single sign-on through CloudAccess:

  - **I do not have an existing account. Create one for me.**
  - **I already have an existing account. These are my credentials:**

When it provisions a new account, CloudAccess determines if a matching user account already exists in the SaaS application. This search occurs whether the provisioning method is automatic or user controlled.

- If a match is found, CloudAccess merges the user with the existing SaaS application account. For more information, see Section 7.4.5, “Matching Criteria for Merging Existing Accounts,” on page 89.
- If no match is found, Cloud creates a new account. For more information, see Section 7.4.4, “CloudAccess Naming Convention for Newly Provisioned Accounts,” on page 88.

Whether CloudAccess merges the user account or creates a new account, the user’s SaaS application password is set to a random value that CloudAccess generates. The user’s authentication to the SaaS application now uses federated single sign-on with SAML 2.0 or WS-Federation. CloudAccess sends information about an authenticated user to the destination application by using an assertion or token. The user does not log in directly to the application. For information about single sign-on for users, see Section 7.3, “Providing Access to Applications for Users,” on page 85.

7.4.3 Samples of Account Creations

The examples in this section describe the experience for automatic account creation and user-controlled provisioning (that is, when the **Prompt users for an existing account before provisioning** is enabled).

Sample experience with the automatic account provisioning:

1. The administrator maps one or more SaaS authorizations to the identity source role (group).
2. (Conditional) If approval is required, the person with the Approval role approves or denies the account creation.
3. CloudAccess searches for an existing, matching account.
4. CloudAccess merges an existing matching account. If CloudAccess finds a matching account, CloudAccess merges the existing account with the new account it creates for the user. For example, the user’s mail attribute in the identity source matches a user name in the Salesforce domain.

   Or

   CloudAccess provisions a new account. If CloudAccess does not find a matching account, CloudAccess creates a new account for the user in the SaaS application, following the naming conventions in Table 7-1 on page 89.
5. Users log in to CloudAccess with their corporate credentials, and CloudAccess authenticates them against the identity sources.
6. CloudAccess presents users with appmarks for the SaaS applications they are entitled to access.

7. Users click the appmark for the entitled SaaS application.
8. CloudAccess uses single sign-on to authenticate the users to the SaaS application.

Sample experience with user-controlled account provisioning:

1. The administrator selects the **Prompt users for an existing account before provisioning** option when configuring the connector for the SaaS application.
2. The administrator maps one or more SaaS authorizations to the identity source role (group) and grants approval, if required.
3. (Conditional) The person with the Approval role approves or denies the account creation.
4. Users log in to CloudAccess with their corporate credentials, and CloudAccess authenticates them against the identity sources.
5. CloudAccess presents users with appmarks for the SaaS applications they are entitled to access.
6. Users click the appmark for the entitled SaaS application.
7. CloudAccess presents two options to users:
   - **I do not have an existing account. Create one for me.**
   - **I already have an existing account. These are my credentials:**
8. Users select **I do not have an existing account. Create one for me.**

   or

   Users select **I already have an existing account. These are my credentials**, then specify their credentials for the existing SaaS account.
9. Regardless of the option the user selects, CloudAccess searches for an existing, matching account.
10. CloudAccess merges an existing matching account. If CloudAccess finds a matching account, CloudAccess merges the existing account with the new account it creates for the user. For example, the user's mail attribute in the identity source matches a user name in the Salesforce domain.

    Or

    CloudAccess provisions a new account. If CloudAccess does not find a matching account, CloudAccess creates a new account for the user in the SaaS application, following the naming conventions in Table 7-1 on page 89.
11. CloudAccess uses single sign-on to authenticate the users to the SaaS application.

7.4.4 CloudAccess Naming Convention for Newly Provisioned Accounts

CloudAccess contains a defined naming convention for creating the user accounts in the SaaS applications.

Table 7-1 CloudAccess Naming Convention

Identity Source	Connector for Google Apps	Connector for Microsoft Office 365	Connector for Salesforce
Active Directory	sAMAccountName	sAMAccountName@*Federated Domain Name*	sAMAccountName@*Salesforce Domain Name*
eDirectory	CN	CN@*Federated Domain Name*	CN@*Salesforce Domain Name*
JDBC	CN	CN@*Federated Domain Name*	CN@*Salesforce Domain Name*

As shown in Table 7-1, the user name attribute for Office 365 and Salesforce are in the form of an email address. For Office 365, CloudAccess creates a unique user name based on the sAMAccountName or CN of the user object in the identity source prepended to the federated domain name configured for the Organization in the Office 365 account. For Salesforce, CloudAccess creates a unique user name based on the sAMAccountName or CN of the user object in the identity source prepended to the domain name configured in the Company Profile at the Salesforce account.

7.4.5 Matching Criteria for Merging Existing Accounts

The following sections define the matching criteria of the connectors that provision users:

- "Matching Criteria for the Connector for Google Apps" on page 89
- "Matching Criteria for the Connector for Office 365" on page 90
- "Matching Criteria for the Connector for Salesforce" on page 90

Matching Criteria for the Connector for Google Apps

Table 7-2 contains the matching criteria for the connector for Google Apps. CloudAccess compares the Google Apps email attribute value to the listed identity source attribute value. If there is a match, CloudAccess merges the accounts. If there is no match, CloudAccess creates a new account in Google Apps. The Display Name is the user-friendly name for the attribute parameter that it represents.

Table 7-2 Google Apps Matching Criteria

Source	Display Name	Attribute Name
Google Apps	Email	Username
Active Directory	User logon name	sAMAccountName
eDirectory	Username	CN
JDBC	Username	CN

Matching Criteria for the Connector for Office 365

Table 7-3 contains the matching criteria for the connector for Office 365. CloudAccess compares the Office 365 `userPrincipalName` attribute value with the value in the identity source attribute plus the @ sign plus the Office 365 federated domain name. If the values match, CloudAccess merges the accounts. If there is no match, CloudAccess creates a new account in Office 365. The Display Name is the user-friendly name for the attribute parameter that it represents.

Table 7-3 *Office 365 Matching Criteria*

Source	Display Name	Attribute Name
Office 365	User name	userPrincipalName (upn)
Active Directory	User logon name@*Federated Domain Name*	sAMAccountName@*Federated Domain Name*
eDirectory	Username@*Federated Domain Name*	CN@*Federated Domain Name*
JDBC	Username@*Federated Domain Name*	CN@*Federated Domain Name*

Matching Criteria for the Connector for Salesforce

Table 7-4 contains the matching criteria for the connector for Salesforce. CloudAccess matches on three different attributes in a priority order. If CloudAccess does not find a match for the value in the first attribute, it performs a search in the value of the second attribute, and if it does not find a match, it performs the search for the value in the third attribute. The Display Name is the user-friendly name for the attribute parameter that it represents.

Table 7-4 *Salesforce Matching Criteria*

Source	Display Name	Attribute Name
First Priority		
Salesforce	Federation identifier	
Active Directory		objectGUID
		employeeID
eDirectory		GUID
		workforceID
JDBC		
Second Priority		
Salesforce	Username	Username
Active Directory	User logon name@*Salesforce Domain Name*	sAMAccountName@*Salesforce Domain Name*
eDirectory	Username@*Salesforce Domain Name*	CN@*Salesforce Domain Name*
JDBC	Username@*Salesforce Domain Name*	CN@*Salesforce Domain Name*
Third Priority		

Source	Display Name	Attribute Name
Salesforce	Username	Username
Active Directory	E-mail	Mail
eDirectory	E-mail Address	Internet EMail Address
JDBC		

7.5 Configuring Appmarks for Connectors

Appmarks are essentially bookmarks for applications. After you configure a connector for an application, you configure one or more appmarks to enable users to access the application in different ways. After a user logs in to CloudAccess, users see the appmarks on the landing page that they are entitled to see, according to the application settings for public access or policy mappings for the application to identity source roles (groups).

You can configure appmarks for any proxy connector, SaaS connector, or SSO connector. You can even configure multiple appmarks for the same connector. For example, you might want to have several appmarks for the various Office 365 applications so users can easily identify them. The connector for Google Apps includes default appmarks for Calendar, Drive, and Mail applications. You can copy an existing appmark to create a new one.

When you configure an appmark, you specify whether you want the application to launch in a desktop browser or on a supported mobile device, or both. If you configure a single appmark to display in both a desktop browser and on a mobile device, the appmark will have the same name, but you can customize the icons so they are different. Appmarks offer significant flexibility, enabling you to customize your users' experience using different view options and variables.

When you configure a new appmark to display on a mobile device, after the appliance is finished applying your change, the user must do a refresh on the mobile device before the appmark appears. To do a refresh, the user does the standard "pull-to-refresh" action on the Applications page in the MobileAccess app. (This action is used in mail and other common applications on the mobile device.)

NOTE: Appmarks for proxy and SSO connectors have no access control associated with them. If users know how to get to a service, they can access the service. Appmarks just add convenience to the user experience.

Use the information in the following sections to help you understand and configure appmarks:

- Section 7.5.1, "Understanding Appmark Options," on page 92
- Section 7.5.2, "Mobile Device Workflow using Safari or Chrome," on page 93
- Section 7.5.3, "Mobile Device Workflow with Internal Viewer," on page 94
- Section 7.5.4, "Mobile Device Workflow from Bookmarks," on page 94
- Section 7.5.5, "Configuring an Appmark for the Desktop Browser or Mobile Device," on page 94
- Section 7.5.6, "Creating Multiple Appmarks for an Application," on page 95
- Section 7.5.7, "Using Appmark Variables," on page 96
- Section 7.5.8, "Policy Mapping for Non-Public Appmarks," on page 96

7.5.1 Understanding Appmark Options

You configure appmarks on the Appmarks tab in the configuration window for the connector. On the Appmarks tab next to the name of the appmark in the blue bar are several icons for renaming, copying, disabling, or deleting the appmark. Use the mouseover text to identify the icon you want to use. You can view and edit appmark configuration options by clicking the blue bar or the plus sign (+) icon. The following appmark options are available:

Reset

This check box restores the Appmarks tab to the default settings for the connector. Consider using this option if you have configured custom connectors that are not working as expected. Click **OK** and apply the changes to the appliance to see the default appmark settings.

Name

The display name for the appmark. If you want different display names for the appmark on the desktop browser page and on mobile devices, you should create a copy of the appmark and change the name. For more information, see Section 7.5.6, “Creating Multiple Appmarks for an Application,” on page 95.

Public

This option is available only for appmarks configured for Simple Proxy, Bookmark, OAuth2 Resources, and SSO-only type connectors. Public access is disabled by default for all connectors except connectors for Basic SSO. If you select the **Public** option, all users can see and use the appmark. If you deselect the **Public** option, no users can see the appmark until it is mapped to desired identity source roles (groups) in Policy Mapping.

Desktop browser

Enables the appmark to be visible on the CloudAccess landing page.

Initiate login at

Specifies whether the URL of the appmark on the landing page is the identity provider-initiated type or the service provider-initiated type. This option appears only for the full provisioning connectors (Google Apps, Salesforce, and Office 365) and the SSO-only connectors, such as Box or Accellion.

URL

The URL that is to be used for the appmark. There are some replacement values that you can use. For more information, see Section 7.5.7, “Using Appmark Variables,” on page 96.

Icon

The icon that appears on the landing page. Within the same appmark, you can use different icons for the landing page and for mobile devices. You can use a different custom icon for each connector to improve their usability for users.

iOS devices

Enables the appmark to be visible on supported iOS mobile devices in the MobileAccess app on the Applications page.

Android devices

Enables the appmark to be visible on supported Android mobile devices in the MobileAccess app on the Applications page.

Launch with

Specifies how to launch the application on the mobile device. Options include the following:

- **Safari:** When the user opens the MobileAccess app on the mobile device and taps the appmark, the MobileAccess app launches Safari and directs it to the application.
- **Chrome:** When the user opens the MobileAccess app on the mobile device and taps the appmark, the MobileAccess app launches Chrome and directs it to the application. If Chrome is not installed on the mobile device, the user is taken to the App Store to install it.
- **Internal viewer:** When the user opens the MobileAccess app on the mobile device and taps the appmark, the MobileAccess app opens an embedded HTML viewer and directs it to the application. This view is similar to the Safari and Chrome options, except that the user does not have to leave the MobileAccess window. The application opens within the MobileAccess app window, and the user can tap the app name (as defined by the administrator when configuring the tool in the appliance) on the navigation bar in the top left corner of the screen to go back to the app home page and easily switch to another protected resource.
- **Native application:** Use this option specifically for mobile apps. When the user opens the MobileAccess app on the mobile device and taps the appmark, MobileAccess opens the mobile app itself.

Launch URL

Use for the **Native application** option. This is the URL such as `fb://profile` that will launch another application installed on the device.

App installer URL

(Optional) You can use this option if you selected the **Native application** option. This is the URL to install the application if it is missing on the mobile device.

URL

The URL that is to be used for the appmark. This can be different from the desktop URL if there is a mobile-specific version of the page.

Icon

The icon that represents the application in the MobileAccess app. Appmark icons for mobile devices should be in `.png` file format and ideally 72 x 72 pixels to ensure they display correctly. Square icons size well on mobile devices. Each icon should convey a good visual image of the application it represents.

7.5.2 Mobile Device Workflow using Safari or Chrome

When you select **Safari** or **Chrome** from the appmark **Launch with** list, MobileAccess opens the application in a new tab in the browser by using the MobileAccess proxy.

The browser workflow on the mobile device is as follows:

1. The end user opens the MobileAccess app on the mobile device.
2. (Conditional) If it is configured, the user is prompted for and enters an application PIN.
3. The user sees a list of protected resources and selects a protected resource.
4. The MobileAccess app starts Safari or Chrome and directs it to the protected resource via the MobileAccess proxy by opening a new tab in the browser.

5. The end user is allowed access to the protected resource.
6. In Google Chrome, the user can tap the button in the top left of the navigation bar to close the current tab and return to the MobileAccess app.

7.5.3 Mobile Device Workflow with Internal Viewer

When you select **Internal viewer** from the appmark **Launch with** list, MobileAccess opens an embedded HTML viewer and directs it to the protected resource by using the MobileAccess proxy.

The workflow on the mobile device is as follows:

1. The end user opens the MobileAccess app on the mobile device.
2. (Conditional) If it is configured, the user is prompted for and enters an application PIN.
3. The user sees a list of protected resources and selects a protected resource.
4. The MobileAccess app opens an embedded HTML viewer and directs it to the protected resource using the MobileAccess proxy.
5. The end user is allowed access to the protected resource.

7.5.4 Mobile Device Workflow from Bookmarks

When a user opens a protected bookmarked application in a Safari browser, MobileAccess prompts the user for the application PIN, then allows the user to access the bookmarked application.

The workflow using bookmarks on the mobile device is as follows:

1. The end user opens Safari on the mobile device.
2. The end user selects a bookmark that points to a URL protected by MobileAccess (i.e., a protected resource).
3. The end user is redirected to the MobileAccess app.
4. (Conditional) If it is configured, the user is prompted for and enters an application PIN.
5. The end user is redirected back to Safari and the bookmarked URL (protected resource).
6. The end user is seamlessly allowed access to that bookmarked application.

7.5.5 Configuring an Appmark for the Desktop Browser or Mobile Device

After you have configured a connector for a proxy, SaaS, or SSO application, you can configure an appmark to simplify access to that application from the user's landing page or from a mobile device, or both.

To configure an appmark:

1 Log in with an appliance administrator account to the Admin page at

`https://appliance_dns_name/appliance/index.html`

2 (Conditional) If you have not already configured the connector for the application, drag it from the **Applications** palette to the **Applications** panel.

3 Click the configured connector on the **Applications** panel and click **Configure**.

4 (Conditional) If you have not already configured the connector, provide the appropriate information on the **Configuration** tab. The required information varies depending on the connector.

5 Click the **Appmarks** tab.

6 Click the plus (**+**) sign next to the default created appmark.

7 (Conditional) Select the **Public** check box if you want the appmark to appear for all users, regardless of their entitlement to the application.

8 (Conditional) If you want the appmark to be available on the user's landing page, select the **Desktop browser** check box and complete the following steps:

   8a (Conditional) If it is applicable to the connector, select the appropriate option from the **Initiate login at** list.

   8b Leave the default value in the **URL** field.

   8c (Optional) If you want to provide your own icon for the appmark, click the **X** on the **Icon** line to delete the default icon. Then browse to and select a `.png` file to represent the application on the browser's landing page.

9 (Conditional) If you want the appmark to be available on the user's mobile device, select the **iOS devices** or **Android devices** check box and complete the following steps.

   9a Select an option from the **Launch with** list to specify how you want users to access the application on their mobile device. For more information about the available options, see Section 7.5.1, "Understanding Appmark Options," on page 92.

   9b (Optional) If you want to provide your own icon for the appmark, click the **X** on the **Icon** line to delete the default icon. Then browse to and select a `.png` file to represent the application on the mobile device. You can use different icons for the landing page and mobile devices.

10 Click **OK**, then click **Apply**.

The appliance reconfigures with the new change. After this process has completed, users who enter the appliance URL are redirected to a login page. They enter their user name and password and are presented with a landing page containing the appmark icon that links to the application.

7.5.6 Creating Multiple Appmarks for an Application

Application connectors can have multiple appmarks. For example, you might create several appmarks for different Office 365 or Google Apps applications. You can create a new appmark from scratch, or you can copy an existing appmark to save time, especially if you want to create several appmarks and just change one or two options on each one. This procedure assumes you have already configured the connector.

To create a new appmark for a connector:

1 Log in with an appliance administrator account to the Admin page at

   `https://appliance_dns_name/appliance/index.html`

2 Click the configured connector on the **Applications** panel, then click **Configure**.

3 Click the **Appmarks** tab, then do one of the following:

   - Click **New**
   - Click the **Copy** icon next to the existing appmark name

4 (Conditional) If you are copying an existing appmark, the **Name** field is pre-populated with `COPY_$(DisplayName)`. You have several options:

   - You can accept this default name. (However, note that "COPY_" will be part of the name.)

- You can change the display name by manually editing the text.
- You can edit the display name by selecting from available variables. Type ${ at the end of the field, then select a variable from the list. For more information about the available variables, see Section 7.5.7, “Using Appmark Variables,” on page 96.

5 Specify whether the application should be accessible from a desktop browser or a mobile device, or both, and complete the appropriate fields. For more information about available options, see Section 7.5.1, “Understanding Appmark Options,” on page 92.

6 Click **OK**, then click **Apply** to update the appliance.

7.5.7 Using Appmark Variables

Each connector has different configuration settings and variables, and some appmarks need to contain information from the connector configuration to be useful. When you configure a connector, the Appmarks tab is automatically populated with one or more default appmarks, depending on the connector. The default settings contain some variables in the URL field.

You can use the variables that are available for a connector in the **Name** and **URL** fields if they are of the string type and have a value provided. To insert a variable, type `${` to display the available variables. Use the mouse or press the up/down arrow keys to select a variable. When you press the down arrow key, an additional box shows the resolved value. Press the up arrow key to close the resolved variables box. Some variables may not be resolvable until after you apply your changes on the appliance.

7.5.8 Policy Mapping for Non-Public Appmarks

Appmarks for proxy and SSO applications are intended only for display and convenience. They are not connected to any authorization policy or access control list (ACL). The SSO and proxy appmark URLs are still available to be used by anyone who knows the link in the URL field. However, selecting or deselecting the **Public** option when configuring an appmark determines whether the appmark actually appears for the users in a group. If you deselect the **Public** check box, the appmark is not available for users until you map the appmark to one or more groups in your configured identity source. After mapping is completed, users in those mapped groups can see the appmark on the landing page or mobile device.

The following procedure assumes that you have already configured an appmark and applied the change on the appliance.

To map an appmark to a group in your identity source:

1 Switch to the Policy page of the administration console.

2 On the left side, locate the identity source that has the desired group (listed as Role Name) from the list.

3 On the right side, select **Other Applications** from the list.

4 Select the Authorization Name of the appmark and drag it to a Role Name.

5 In the mapping window, there are no approvals for appmarks because there is no account provisioning in this process. Users who are included in the group are automatically approved. Click **OK** to continue.

Now when users who are in the mapped group do a refresh in the MobileAccess app or access the landing page, they see the new appmark icon. Users who are not in the mapped group do not see the icon.

7.6 Connector for Google Apps for Business

The connector for Google Apps for Business provides automated provisioning of accounts from the identity sources to Google Apps. The connector also provides federated single sign-on access to Google Apps through CloudAccess. The connector allows CloudAccess to authenticate a user against your identity sources and to share this authentication with Google Apps in order to establish the user's session.

CloudAccess includes this connector with the appliance. The connector appears automatically on the **Applications** palette of the Admin page.

Use the information in the following sections to configure a connector for Google Apps:

- Section 7.6.1, "Connector Requirements," on page 97
- Section 7.6.2, "Understanding Google Apps Provisioning," on page 97
- Section 7.6.3, "Creating a Google Service Account," on page 98
- Section 7.6.4, "Configuring the Connector for Google Apps," on page 99
- Section 7.6.5, "Configuring Multiple Connectors for Google Apps," on page 100
- Section 7.6.6, "Updating the Connector for Google Apps," on page 100

7.6.1 Connector Requirements

The connector for Google Apps creates a federated connection between CloudAccess and Google Apps for Business. The connector uses OAuth 2.0 to create the federation.

The connector for Google Apps supports account provisioning only for users in Active Directory, eDirectory, and JDBC identity sources. For more information, see Section 7.4.1, "Requirements for Provisioning," on page 86.

Verify that you meet the following requirements before you configure a connector for Google Apps:

- ❒ An understanding of identity federation using the OAuth 2.0 protocol
- ❒ A CloudAccess system, installed and configured
- ❒ A valid Google Apps for Business account
- ❒ An administrative account and password

7.6.2 Understanding Google Apps Provisioning

Using the Google Apps Admin console, you can configure your Google domain with an organizational structure. Using the same console, you can also assign or revoke Google Apps services such as Mail, Calendar, or Drive to or from specific organizational units within that organizational structure. As a result, user access to Google Apps services is controlled based on the user's location within the organizational structure.

CloudAccess provides support for provisioning users to specific organizational units previously configured in the Google Apps domain. After you have configured the Google Apps organizational structure and services using the Google Apps Admin console, you can configure CloudAccess to provision users to specific locations within that organizational structure.

By default, the connector for Google Apps places newly provisioned users into the top-level organization of your Google Apps domain. For example, if your Google Apps domain is mygmail.com, the connector places users in the mygmail.com organization. If you want all newly provisioned users to be placed in a sub-organization that you have created in your Google Apps domain, you can specify this organizational unit as the default when you configure the connector.

Instead of a default organizational unit, users can be provisioned to a specific organizational unit based on mappings you create on the CloudAccess Policy page. On the Policy page, the Google Apps organizational units are shown as User Placement type Authorizations. Mapping a User Placement overrides any default organizational unit you specify in the connector configuration.

7.6.3 Creating a Google Service Account

The connector for Google Apps uses an OAuth 2.0 federated connection to provision the user accounts from the identity source to Google Apps. In order to create this federated connection, you must create a Google service account. For more information about this process, see "Using OAuth 2.0 for Server to Server Applications" (https://developers.google.com/accounts/docs/OAuth2ServiceAccount).

You must create this service account before you can configure the connector for Google Apps. To create the service account, follow the instructions on the Google developer site: "Creating a service account" (https://developers.google.com/accounts/docs/OAuth2ServiceAccount#creatinganaccount).

When you create the service account, enable the following API:

- Admin SDK

In addition, when you create the service account, record the following information. You will need this information when you configure the connector for Google Apps:

- Service account's email address
- Path to the P12 private key file

Finally, you must delegate domain-wide authority to the service account. This step grants the service account access to the Google resources that CloudAccess needs to access to provision the user accounts and allow single sign-on. To grant domain-wide authority, follow the instructions on the Google developer site: "Delegating domain-wide authority to the service account" (https://developers.google.com/accounts/docs/OAuth2ServiceAccount#delegatingauthority).

Add the following Google scopes in a comma-separated format to grant the correct authorizations:

- `https://www.googleapis.com/auth/admin.directory.group`
- `https://www.googleapis.com/auth/admin.directory.group.member`
- `https://www.googleapis.com/auth/admin.directory.orgunit`
- `https://www.googleapis.com/auth/admin.directory.user`

NOTE: The format of the Google scopes is a comma-delimited list without any hard returns or line breaks. Ensure that when you copy these Google scopes, there are no hard returns or line breaks in the text.

7.6.4 Configuring the Connector for Google Apps

The connector for Google Apps provides user account provisioning and single sign-on access to Google Apps domains. After users log in to CloudAccess, OAuth authentication is used to automatically authenticate (single sign-on) users to Google Apps. Each cluster can support multiple instances of the connector.

To configure the connector:

1. (Conditional) If you want to enable user access to specific Google applications, complete the following steps:
   - **1a** In the Google Apps Admin console, create one or more organizational structures underneath the top level container of your Google Apps domain.
   - **1b** Click the **Google Apps** > **Services** menu option and enable or disable each available application for each selected organization.
2. Log in as an administrator to the CloudAccess administration console:

   `https://appliance_dns_name/appliance/index.html`
3. Drag and drop the connector for Google Apps from the **Applications** palette to the **Applications** panel.

   The Configuration window opens automatically for the initial configuration. To view or reconfigure the settings later, click the connector icon, then click **Configure**.
4. Provide a unique display name for the connector to appear on the Admin and landing pages, and also provide the administrator logon credentials and domain for the Google Apps for Business account.
5. (Conditional) Select the **Automatically configure SSO settings** option if you want CloudAccess to configure the single sign-on parameters for your Google domain. Otherwise, you must manually configure the parameters at Google Apps.

   or

   Click **Federation Instructions**. Read and follow the instructions provided to configure the connector for Google Apps to allow single sign-on for users.
6. (Conditional) Select **Prompt users for an existing Google Apps account before provisioning** if you want to give users control of when their accounts are provisioned. For more information, see Section 7.6.2, “Understanding Google Apps Provisioning,” on page 97.
7. (Conditional) If you want to specify a default organizational unit for newly provisioned users, expand **Advanced Options** and enter the path to the organization in the **Default OrgUnit** field.

   NOTE: You can specify a sub-organization at any level in your Google Apps organizational structure, using forward slashes, as long as you have set up that structure. For example, `mygoogle.com/employees/fulltime/salary`. If you leave this field blank, the connector places newly provisioned users into the top-level organization of the Google Apps domain.

   ---
8. Click **OK**.
9. On the Admin page, click **Apply** to commit the changes to the appliance.

   When the configuration changes have been applied on each node of the CloudAccess cluster, the application is available to users.
10. (Optional) To provide users with access to Google Apps Mail from supported mobile devices, click the configured connector on the **Applications** panel, then click **Enable email proxy**.

11 (Optional) Modify default appmarks or configure new appmarks to specify how users should access the Google Apps applications. By default, the connector includes three appmarks that are configured for the Calendar, Mail, and Drive applications.

   For more information, see Section 7.5, “Configuring Appmarks for Connectors,” on page 91.

12 Add users to the appropriate identity source group to trigger user account provisioning to Google Apps.

13 Perform policy mapping to specify entitlements for identity source groups.

   For more information, see “Mapping Authorizations” in the *NetIQ® CloudAccess and MobileAccess Installation and Configuration Guide*.

14 (Conditional) If required, grant approvals for mapped authorizations.

User accounts that have been provisioned to Google Apps using CloudAccess must authenticate through CloudAccess. Direct logins to Google Apps are not allowed.

7.6.5 Configuring Multiple Connectors for Google Apps

CloudAccess can support Google Apps domains by using multiple instances of the connector for Google Apps. Each connector instance must be configured with the unique credentials and domain information of the Google Apps domain that it serves.

NOTE: The **Enable email proxy** option is global across all instances of the Google Apps connector. (The option is either enabled or disabled for all instances.)

7.6.6 Updating the Connector for Google Apps

The connector for Google Apps has changed in this release. When you update the CloudAccess appliance to version 2.2, the connector for Google Apps stops running and turns red. The updated connector for Google Apps contains additional fields you must configure before the connector can run again.

To update the connector for Google Apps:

1 Perform an update of the appliance.

   For more information, see Updating the Appliance in the *NetIQ® CloudAccess and MobileAccess Installation and Configuration Guide*.

2 Create a Google service account.

   For more information, see Section 7.6.3, “Creating a Google Service Account,” on page 98.

3 Log in to the CloudAccess administration console at `https://appliance_dns_name/appliance/index.html`.

4 Click the connector for Google Apps, then click **Configure**.

5 Add the Google service account’s email address and the path to the P12 private key file.

6 Click **OK**, then click **Apply** to save the changes.

   The new connector is now green and running.

7.7 Connector for Microsoft Office 365 (SAML 2.0 or WS-Federation)

The connector for Microsoft Office 365 provides automated provisioning of accounts from the identity sources to Office 365. The connector also provides federated single sign-on access to Office 365 through CloudAccess. The connector allows CloudAccess to authenticate a user against your identity sources and to share this authentication with Office 365 in order to establish the user's session. The connector supports the SAML 2.0 protocol or the WS-Federation protocol for federated SSO.

With WS-Federation, the connector supports federated single sign-on natively from a Microsoft Lync client or a Lync mobile app for iOS and Android devices. The user must install and configure the NetIQ MobileAccess app and the Lync app to allow this interaction on a mobile device. The user signs in on the Lync login page as usual. The redirection to CloudAccess for authentication and service access is transparent for the user.

CloudAccess includes this connector with the appliance. However, the connector does not appear automatically in the Applications palette of the Admin page. You must run the connector for Office 365 installer on your Windows Management Server in order to connect the web application to CloudAccess.

IMPORTANT: The connector for Office 365 is a CloudAccess-only feature and is not included in the MobileAccess-only license. For more information, see "Understanding Product Licensing" in the *NetIQ® CloudAccess and MobileAccess Installation and Configuration Guide*.

Each cluster supports multiple instances of the connector for Office 365, but each connector must serve a unique domain.

- Section 7.7.1, "How the Connector for Office 365 Works," on page 101
- Section 7.7.2, "Connector Requirements," on page 103
- Section 7.7.3, "Installing the Connector for Office 365," on page 105
- Section 7.7.4, "Validating the Connector for Office 365," on page 106
- Section 7.7.5, "Configuring Appmarks for Office 365 Applications," on page 106
- Section 7.7.6, "Changing the Configuration of the Connector," on page 106
- Section 7.7.7, "Changing the Name of an Office 365 Security Group," on page 106
- Section 7.7.8, "Upgrading the Connector from SAML 2.0 to WS-Federation," on page 107
- Section 7.7.9, "Uninstalling the Connector for Office 365," on page 107
- Section 7.7.10, "Installing Multiple Connectors for Office 365," on page 107

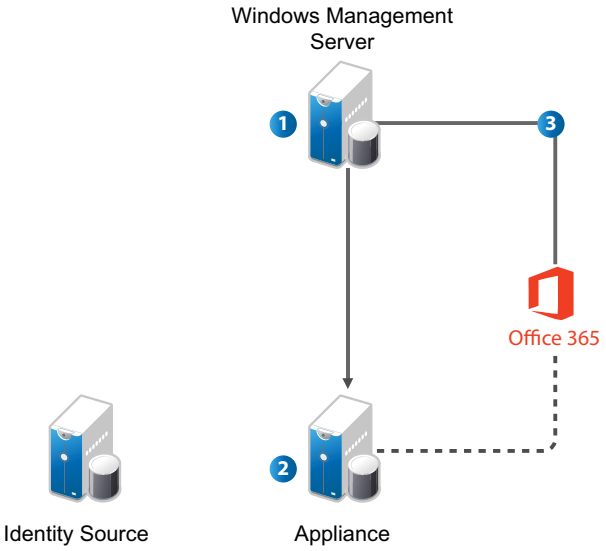
7.7.1 How the Connector for Office 365 Works

Before you install the connector for Office 365, review the following illustrations to help you understand how the connector works with CloudAccess.

- "Setup and Configuration" on page 102
- "User Provisioning" on page 102
- "User Login to Office 365" on page 103

Setup and Configuration

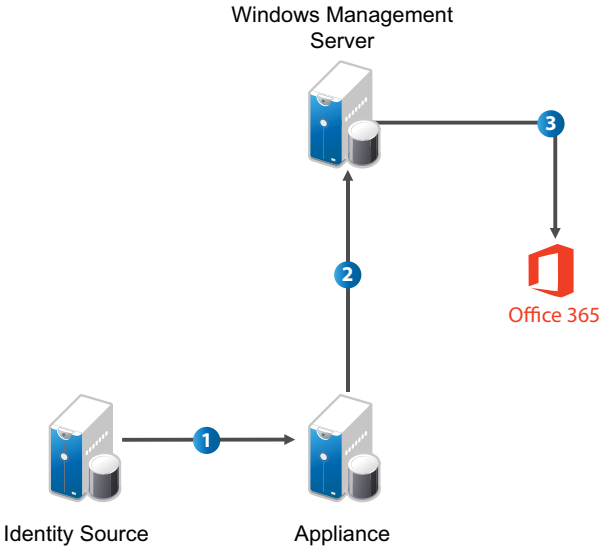
The following figure illustrates the basic setup and configuration steps.



1. Run the connector for Office 365 installer using the `.msi` file. For more information, see Section 7.7.3, “Installing the Connector for Office 365,” on page 105.
2. Create a trust relationship between the CloudAccess appliance and Office 365.
3. Configure user provisioning to Office 365.

User Provisioning

The following figure illustrates the workflow in provisioning users.

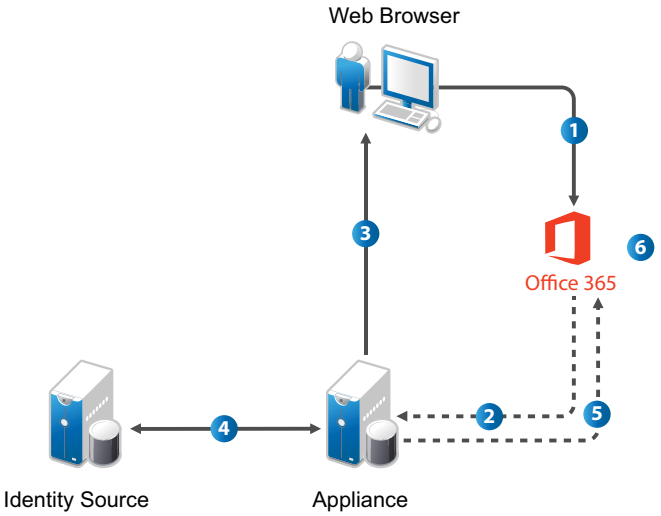


1. The administrator defines a policy to authorize access to Office 365 applications.
2. CloudAccess detects new and updated user information from the identity source.

3. CloudAccess sends user creation, license assignment, update, or deletion requests to the Windows Management Server.
4. The Windows Management Server forwards requests to Office 365 using the Windows Azure Active Directory Module for Windows PowerShell cmdlets.

User Login to Office 365

The following figure illustrates the workflow for users logging in to Office 365 applications.



1. The user attempts to log in to Office 365.
2. The login is redirected to CloudAccess.
3. CloudAccess prompts the user for the user name and password. Or, if Kerberos is configured, CloudAccess performs seamless authentication.
4. CloudAccess verifies the user name and password using the identity sources. Or, if Kerberos is configured, CloudAccess validates the Kerberos token.
5. CloudAccess provides an assertion to Office 365.
6. Office 365 validates the assertion and allows the user access to assigned Office 365 applications.

7.7.2 Connector Requirements

The connector for Office 365 supports account provisioning only for users in Active Directory, eDirectory, and JDBC identity sources. For more information, see Section 7.4.1, “Requirements for Provisioning,” on page 86.

Complete the following steps before you install the connector for Office 365:

- ❒ Identify an existing Office 365 administrative account to use, or create a new administrative account. This administrative user must not belong to the Office 365 domain that CloudAccess will manage.
- ❒ Identify the verified Office 365 domain for which CloudAccess will manage authentication.

  Microsoft does not support subdomains having different federated settings than their parent. To use a subdomain for Office 365, ensure that either you do not use Office 365 with the parent domain, or that both the parent domain and its subdomain have the identical federation settings.

- ❒ Identify a Windows Management Server on which to install the connector. The Windows Management Server that you use for the connector for Office 365 should be a dedicated server that is not used for other web applications. However, the connector does not need to be installed on a domain controller or even need to be part of the domain where the CloudAccess appliance is installed. The Windows server can be a standalone server, as long as it meets the following requirements:
  - Windows Server 2012 R2 or Windows Server 2008 R2 operating system with all available updates installed.
  - Microsoft IIS. Install the IIS component that comes with the Windows Server operating system on your Windows Management Server. Use the default IIS settings. The connector uses HTTPS between the CloudAccess appliance and the Office 365 web application in IIS.
  - Microsoft .NET Framework 4.x. You can download .NET from the .NET downloads (http://www.microsoft.com/net/downloads) web page.
  - Microsoft Online Services Sign-In Assistant 7.x. You can download the Microsoft Online Services Sign-In Assistant software from the following location: Microsoft Online Services Sign-In Assistant for IT Professionals BETA (http://www.microsoft.com/en-us/download/details.aspx?id=39267). Select the `msoidcli_64.msi` file.
  - Windows Azure AD Module for Windows Powershell. You can download the module from the following location: Manage Windows Azure AD using Windows PowerShell (http://technet.microsoft.com/en-us/library/jj151815.aspx#bkmk_installmodule).
- ❒ The Microsoft Lync support is available only if you configure the connector with WS-Federation.
- ❒ (Conditional) If you plan to use the Enhanced Client Profile (ECP), also called *HTTP proxy authentication*, in Microsoft Outlook, or if you plan to use Microsoft Lync, ensure that you configure CloudAccess with the following:
  - A publicly resolvable, publicly accessible IP address. You can use port forwarding to protect your appliance behind your corporate firewall.

    When the user logs in to the Office 365 online portal, the browser handles all of the redirects and name resolution, so you can manually edit entries in the device's `../etc/hosts` files to work around name resolution. However, with ECP and Lync, Office 365 actually sends an authentication request directly to CloudAccess, so its IP address must be publicly accessible.
  - An SSL certificate signed by a trusted certificate authority (CA) such as Verisign, Thawte, Symantec, Digicert, and so on. The certificate common name must match the appliance hostname.

NOTE: CloudAccess also supports ECP for Microsoft Exchange email on mobile devices running Android or iOS. The users must add an Exchange account on their device and enter their Exchange credentials. For more information, see the following Microsoft web pages:

- *Set up email on an Android phone or tablet*
- *Set up email on Apple iPhone, iPad, and iPod Touch*

If you use WS-Federation for the connector for Office 365, CloudAccess also supports ECP for Microsoft Lync. The users must add a Lync account on their mobile device and enter their Lync credentials. For more information, see the following Microsoft web pages:

- *Getting started with Lync 2013 for Android*
- *Getting started with Lync 2013 for iPhone*
- *Getting started with Lync 2013 for iPad*

7.7.3 Installing the Connector for Office 365

You must install the connector for Office 365 on a Windows Management Server.

To configure the server and install the connector:

1. Obtain the credentials for an Office 365 administrative account. For more information, see the Office 365 website
2. Add the federated domain name to Office 365 that will be used for single sign-on with CloudAccess and Office 365, and then validate the ownership. Use the instructions at the following web page: Add your domain name to Office 365.

   NOTE: Microsoft requires that each Office 365 federated domain be configured with a unique issuer ID. Thus, each instance of the connector for Office 365 connects to only a single unique Office 365 federated domain.

3. Verify that the Windows server where you plan to install the connector has the prerequisite software installed. For more information, see Section 7.7.2, “Connector Requirements,” on page 103.
4. As an administrator on the Windows server, perform the following steps. For more information, see the IIS Manager help. Alternatively, you can use an imported server certificate. For more information, see Importing a Server Certificate (http://technet.microsoft.com/en-us/library/cc732785%28v=ws.10%29.aspx).
   - **4a** Create a self-signed certificate in IIS Manager.
   - **4b** Add an HTTPS binding for the Default Web Site using the certificate you created.
   - **4c** Restart the IIS service.
5. As an administrator on the Windows server, download the connector for Office 365 `.zip` file from NetIQ Downloads (https://dl.netiq.com/). Unzip the file and run the Windows `netiq-office365-connector-1.5.1.msi` installer. You will need the following information:
   - DNS name of the CloudAccess appliance.
   - Administrator name and password of the CloudAccess appliance.
   - User name and password for the Office 365 Global administrator account.
   - The federated domain name specified in Step 2. If you get an error during installation, ensure that you selected the correct domain name.

   Alternatively, you can run the connector installer in “silent mode” from the command line as follows:

   ```
   msiexec /i netiq-office365-connector-1.5.1.msi /qb
   AG4CHOSTNAME="DNS_of_appliance" AG4CADMIN="CloudAccess_Admin_username"
   AG4CADMINPASS="CloudAccess_Admin_password"
   O365ADMIN="Office365_Admin_username" O365ADMINPASS="Office365_Admin_password"
   O365FEDDOMAIN="Office365_Federated_Domain_Name" O365USAGELOCATION="US"
   ```

   By default, CloudAccess does not generate an installation log when you install the connector for Office 365. If you want a log of the installation, you must launch the installer from the command line using the following command:

   ```
   msiexec /i netiq-office365-connector-1.5.1.msi /L*V "C:\log\example.log"
   ```

   IMPORTANT: The connector for Office 365 installation location is `c:\NetIQ\Office365Connector`. You cannot change this location.

After you have installed the connector, when you return to the CloudAccess administration console, the connector for Office 365 icon is automatically moved to the Applications panel. No additional configuration is required, but you can configure appmarks for different Office 365 applications, and map users to the appropriate applications to set their entitlements. For more information about configuring appmarks, see Section 7.7.5, “Configuring Appmarks for Office 365 Applications,” on page 106.

7.7.4 Validating the Connector for Office 365

After you have installed the connector, perform the following steps to validate the installation:

1. Verify that the connector for Office 365 appears on the Applications panel of the Admin page of the console.
2. Verify that the Policy Mapping page displays Identity Source Groups on the left side and the connector for Office 365 on the right side.
3. Map one of your groups to the User Authorizations, then verify that CloudAccess provisioned your users.
4. Log in to Office 365 at http://www.office365.com as a provisioned user.
5. Specify the user name of `user@domain` where the domain is the federated domain name specified in Step 2 on page 105.

NOTE: If you have any issues with the connector, check the Windows Event Viewer on the Windows server where the connector is installed. You can view all events for the connector to help troubleshoot those issues. In the Windows Event Viewer, expand **Windows Logs**, then click **Application**.

7.7.5 Configuring Appmarks for Office 365 Applications

By default, the connector for Office 365 includes a single appmark that is configured for the user’s home page. You can modify this default appmark or create additional appmarks as needed. Appmarks that are then mapped in Policy Mapping appear on the landing page or in the MobileAccess app for users to whom you have granted access. For more information about configuring appmarks, see Chapter 7.5, “Configuring Appmarks for Connectors,” on page 91.

NOTE: If you configure appmarks for users to launch Office 365 applications using Safari on mobile devices, you should instruct users to set Safari to never block cookies. Alternatively, consider selecting another **Launch with** option to ensure that users do not experience logout errors. For more information, see “Office 365 Logout Error on Mobile Devices”.

7.7.6 Changing the Configuration of the Connector

If you change the federated domain name, you must reinstall the connector for Office 365. The installation changes the configuration information in the connector. Delete the existing connector, then run through the installation again using the new federated domain name.

7.7.7 Changing the Name of an Office 365 Security Group

CloudAccess tracks the security groups in Office 365 by their group name. If you change the name of a security group in Office 365, the rename appears as a delete and add to CloudAccess. CloudAccess deletes the old group and adds the new group. Any authorizations that are mapped to

the deleted group are also removed. After you rename a security group in Office 365, verify that the group appears in CloudAccess with its new name, then go to the Policy page and remap the authorizations for Office 365 to the group.

7.7.8 Upgrading the Connector from SAML 2.0 to WS-Federation

In CloudAccess 2.1 or later, the connector uses WS-Federation to support single sign-on from a Microsoft Lync client. In previous releases, CloudAccess uses SAML 2.0 for single sign-on and provisioning. You must upgrade an existing connector for Office 365 in order to take advantage of the Lync access capability.

To upgrade the connector for Office 365:

1. Log in to the administration console.
2. On the Admin page in the Applications panel, click the connector for Office 365, then select **Configure**.
3. In the Configuration window, select **Use WS-Federation**.
4. Select **Automatically Configure SSO Settings** in order to let the connector modify the federation settings from SAML to WS-Federation.

   If you do not select this option, you must manually reconfigure the **Federation Settings**.
5. Click **OK**.
6. On the Admin page, click **Apply** to commit the changes to the appliance.
7. Wait until the configuration changes have been applied on each node of the CloudAccess cluster.

7.7.9 Uninstalling the Connector for Office 365

The connector for Office 365 consists of multiple components. To correctly uninstall the connector, log in to the Windows server as an administrator and use the uninstall function in Windows Control Panel.

Using the Control Panel to uninstall the connector deletes the connector from the CloudAccess Admin page. If you just delete the connector for Office 365 icon from the Admin page, all of the components on the Windows server still exist and run. This causes issues in CloudAccess if you need to reinstall the connector, unless you run the connector uninstall from the Windows server before attempting to reinstall a new connector.

Alternatively, you can uninstall the connector by running the following command on the Windows server:

```
msiexec /x netiq-office365-connector-1.5.1.msi /qb
```

7.7.10 Installing Multiple Connectors for Office 365

CloudAccess supports multiple connectors for Office 365. However, each connector must connect to a unique Office 365 domain, and you must install each connector on a separate Windows server.

7.8 Connector for Salesforce (SAML 2.0)

The connector for Salesforce provides automated provisioning of user accounts from the identity sources to Salesforce. The connector also provides federated single sign-on access to Salesforce with SAML 2.0 through CloudAccess. The connector allows CloudAccess to authenticate a user against your identity sources and to share this authentication with Salesforce in order to establish the user's session.

In addition, the connector supports single sign-on natively to a Salesforce mobile app for iOS and Android devices. The user must install and configure the NetIQ MobileAccess app and the Salesforce app to allow this interaction. The user signs in on the Salesforce login page as usual.

CloudAccess includes this connector for Salesforce with the appliance. The connector is located on the **Applications** palette of the Admin page.

Use the information in the following sections to configure a connector for Salesforce:

- Section 7.8.1, "Connector Requirements," on page 108
- Section 7.8.2, "Configuring Salesforce to Trust CloudAccess," on page 109
- Section 7.8.3, "Configuring the Connector for Salesforce," on page 109
- Section 7.8.4, "Configuring Appmarks for Salesforce," on page 111
- Section 7.8.5, "Configuring Multiple Connectors for Salesforce," on page 111
- Section 7.8.6, "Using SSO to Salesforce on Mobile Devices," on page 112
- Section 7.8.7, "Configuring Delegated Authentication in Salesforce," on page 115
- Section 7.8.8, "Configuring the Salesforce Federation Identifier," on page 116

7.8.1 Connector Requirements

The connector for Salesforce supports account provisioning only for users in Active Directory, eDirectory, and JDBC identity sources. For more information, see Section 7.4.1, "Requirements for Provisioning," on page 86.

Verify that you meet the following requirements before you configure the connector for Salesforce:

- ❒ An understanding of identity federation using the SAML 2.0 protocol.

  For more information about SAML, see the OASIS website (https://wiki.oasis-open.org/security/FrontPage).

- ❒ A full or developer Salesforce account with provisioning APIs enabled.

- ❒ Administrator access to the Salesforce account. An understanding of Salesforce and its account management tools are presumed.

- ❒ (Conditional) A security token from Salesforce.

  For more information, see Section 7.8.2, "Configuring Salesforce to Trust CloudAccess," on page 109.

- ❒ The location in the Salesforce administration console where you will configure the SAML 2.0 federation for CloudAccess.

  When you configure the connector, the **Federation Instructions** provide the information that you will need to set up the federation in Salesforce for CloudAccess. This information includes the metadata specific to the appliance; a signing certificate for the appliance; the field values to use; and other guidance.

❒ The metadata file from Salesforce.

You generate and download this file after you configure SAML 2.0 federation for CloudAccess in Salesforce.

7.8.2 Configuring Salesforce to Trust CloudAccess

NetIQ recommends that you configure Salesforce to trust the IP address of the CloudAccess appliance.

To add the CloudAccess IP address as a trusted source to Salesforce:

1. Log in to the Salesforce Admin tools web page.
2. Click **Administration Setup** > **Security Controls** > **Network Access**.
3. Specify the IP address of the CloudAccess appliance.

   or

   If you are in a clustered environment, specify the IP address of the L4 switch.

If you do not configure Salesforce to trust the IP address of the CloudAccess appliance, you must obtain a security token from Salesforce and append the security token to the administrator password specified when you configure the connector for Salesforce.

For example, if your Salesforce administrator account password is `Test1234` and the Salesforce security token is `XyZ`, the **Password** field must contain `Test1234XyZ`.

7.8.3 Configuring the Connector for Salesforce

The phone icon that CloudAccess displays on each configured instance of the connector for Salesforce indicates that Delegated Authentication can be used with Salesforce. All of the configuration required for using Delegated Authentication with the appliance is done at Salesforce. For more information, see Section 7.8.7, “Configuring Delegated Authentication in Salesforce,” on page 115.

You must go back and forth between the CloudAccess Admin page and the Salesforce administration page to configure the connector.

To configure the connector for Salesforce:

1. Do one of the following:
   - Configure Salesforce to trust CloudAccess.
   - Obtain a security token from Salesforce.

   For more information, see Section 7.8.2, “Configuring Salesforce to Trust CloudAccess,” on page 109.
2. (Optional) Log in to Salesforce as the account administrator, then enable and configure Salesforce Delegated Authentication single sign-on for your Salesforce organization.

   For more information, see Section 7.8.7, “Configuring Delegated Authentication in Salesforce,” on page 115.
3. Log in with an appliance administrator account to the CloudAccess administration console at

   `https://appliance_dns_name/appliance/index.html`
4. Drag and drop the connector for Salesforce from the **Applications** palette to the **Applications** panel.

The Configuration window opens automatically for the initial configuration. To view or reconfigure the settings later, click the connector icon, then click **Configure**.

5 Specify a unique display name for the connector to appear on the Admin page.

6 Specify the login credentials for the Salesforce administrator user.

NOTE: If you opted not to have CloudAccess as a trusted source for Salesforce in Step 1, you must append the security token to the Salesforce administrator's password.

---

7 In the **Environment** field, specify whether you have a Production, Development, or Sandbox Salesforce environment. The login URL that is used to verify your Salesforce credentials can be different for each of these environments.

8 Select or deselect **Delegated Authentication single sign-on is disabled in Salesforce**, according to your action for Step 2.

9 (Conditional) If delegated authentication is disabled and if you want to give users control of when their accounts are provisioned, select **Prompt users for an existing Salesforce account before provisioning**.

For more information about account provisioning, see Section 7.4, "How CloudAccess Provisions User Accounts," on page 86.

10 Click **Advanced Settings**, and then specify whether the **Federation attribute** should use a GUID or the user's network identity retrieved from the identity source. The Federation attribute stores the user's Salesforce federation ID.

For more information, see Section 7.8.8, "Configuring the Salesforce Federation Identifier," on page 116.

11 Expand the **Federation Instructions**, then copy and paste the instructions into a text file to use during the Salesforce configuration for single sign-on.

NOTE: You must use a text editor that does not introduce hard returns or additional white space. For example, use Notepad instead of Wordpad.

---

12 Click **OK** to save the configuration so far while you configure Salesforce to work with CloudAccess.

The configuration for the connector for Salesforce is not yet complete.

13 Log in to Salesforce as the account administrator, then configure the SAML 2.0 federation for CloudAccess in the Salesforce administration console.

Use the information from the **Federation Instructions** in Step 11 to complete the setup.

NOTE: When you copy the appliance's signing certificate, ensure that you include all leading and trailing hyphens in the certificate's Begin and End tags.

---

14 After you configure federation for CloudAccess in Salesforce, generate and download the Salesforce metadata file.

15 On the CloudAccess Admin page, click the connector for Salesforce, then click **Configure**.

16 Upload the Salesforce metadata file that you downloaded in Step 14 to the connector for Salesforce.

17 Click the **Appmarks** tab, then review and edit the default settings for the appmark.

For more information, see Section 7.8.4, "Configuring Appmarks for Salesforce," on page 111.

18 Click **OK** to save the configuration.

19 On the Admin page, click **Apply** to commit the changes to the appliance.

20 Wait until the configuration changes have been applied on each node of the CloudAccess cluster.

21 Click **Policy** in the toolbar, then perform policy mapping to specify entitlements for identity source roles (groups).

For more information, see "Mapping Authorizations" in the *NetIQ® CloudAccess and MobileAccess Installation and Configuration Guide*.

22 After you complete the configuration, users can log in through CloudAccess to single sign-on to Salesforce. The CloudAccess login page URL is:

```
https://appliance_dns_name
```

For information about single sign-on through the Salesforce mobile app, see Section 7.8.6, "Using SSO to Salesforce on Mobile Devices," on page 112.

7.8.4 Configuring Appmarks for Salesforce

By default, the connector for Salesforce includes a single appmark that is configured for the user's landing page. You can configure the appmark for the desktop browser and supported mobile devices. You can modify this default appmark or create additional appmarks as needed.

NOTE: To enable single sign-on to the mobile Salesforce app, you can select **Native application** from the list of **Launch with** options on the **Appmarks** tab. However, this is not a requirement.

Appmarks that are then mapped in Policy Mapping appear on the landing page and, if configured, in the MobileAccess app for entitled users. For more information, see Section 7.5, "Configuring Appmarks for Connectors," on page 91.

CloudAccess automatically updates users' Salesforce profiles when their policy mapping changes. Salesforce enforces the following restrictions:

- Salesforce updates the user's profile to the new profile only if licenses are available. The connector driver log shows errors if licenses are not available.
- Salesforce will not move a user from a paid license to a free license. Salesforce also enforces this restriction in its administration console. In order to free a license for a paid account, you must de-activate the user.

7.8.5 Configuring Multiple Connectors for Salesforce

Each cluster supports multiple connectors for Salesforce. If you want to configure more than one instance of the connector for Salesforce, each Salesforce account must be configured with a unique URL. Configuring the URL requires Salesforce assistance, and it often takes at least a day to complete.

To configure the Salesforce URL:

1 Log in to the Salesforce administration web page.

2 Click **Administration Setup** > **Domain Management** > **My Domain**.

3 Provide a unique subdomain name for your organization and click **Check Availability**.

4 If the subdomain you specified is available, select the check box to indicate that you agree to the terms and conditions, then click **Register Domain**.

5 Wait for Salesforce to register your domains. This process takes time.

After the registration is complete, Salesforce provides you with a URL that supports SP-initiated logins and is similar to the following:

```
https://<custom_name>.mysalesforce.com
```

7.8.6 Using SSO to Salesforce on Mobile Devices

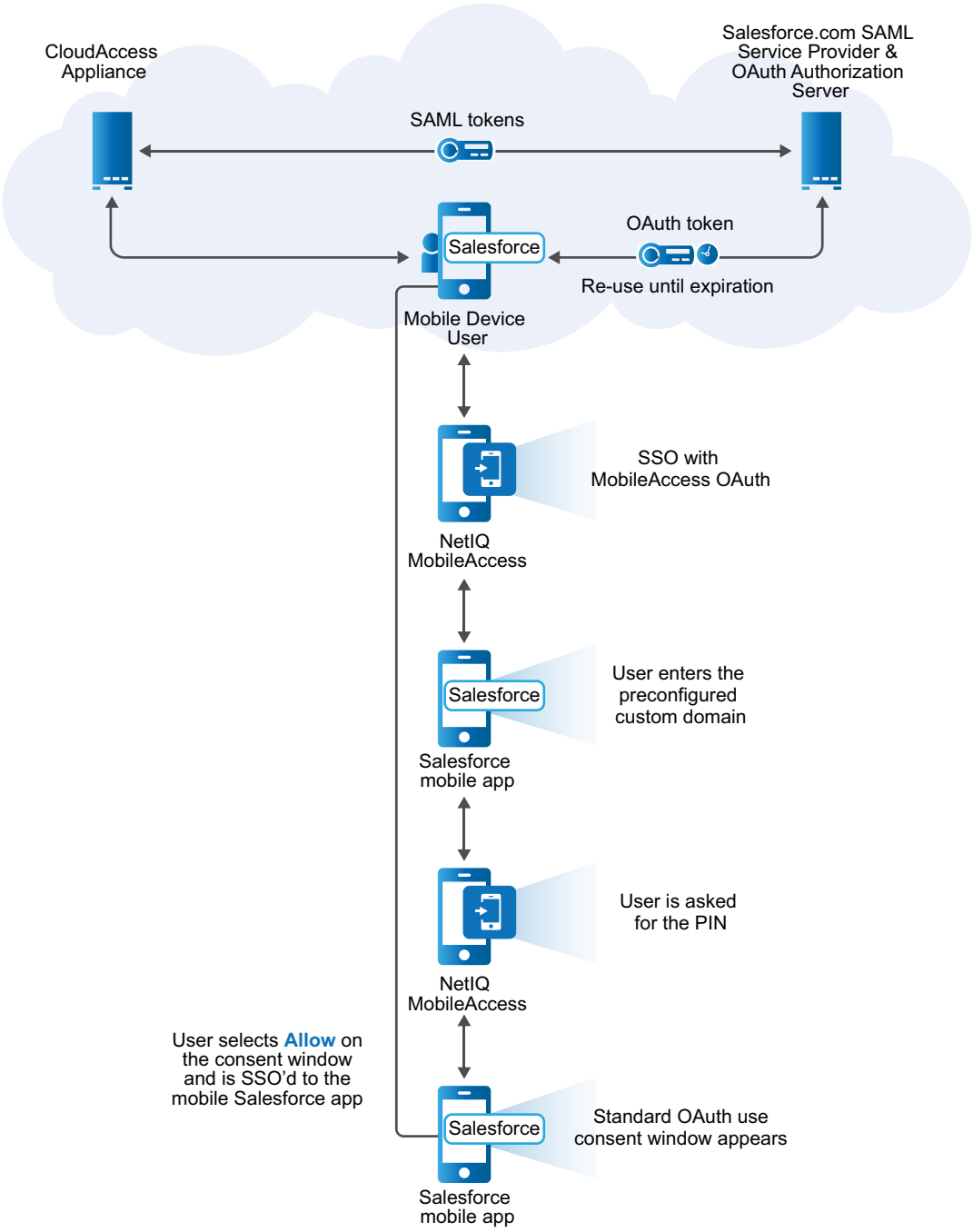
Salesforce now offers a combination of OAuth and SAML functionality to provide seamless SSO facilities not only for web browsers, but also desktop and mobile applications. By using OAuth to enable users to connect applications to their accounts, and leveraging SAML for the authentication of that connection, the single sign-on integration that was previously applicable only for the web browser can now service mobile applications.

The CloudAccess connector for Salesforce provides the necessary protocols and interfaces to the SAML and OAuth features of Salesforce. Single or multi-factor authentication can be used. The mobile device provides a standard browser interface to end users by using HTML. SSL/TLS is used for all appliance connections to protect user credentials and tokens.

Understanding the Mobile SSO Process

The following illustration provides a high-level overview of the mobile SSO to Salesforce process.

Figure 7-3 *SSO to Mobile Salesforce Process*



From the CloudAccess administrator's perspective, the process is as follows:

1. The CloudAccess administrator sets up a SAML trust relationship between the CloudAccess appliance and the Salesforce service to provide user authentication and SAML tokens.
2. The end user authenticates with the CloudAccess appliance to obtain a SAML token and send it to the Salesforce.com server.

3. The Salesforce.com server accepts a valid SAML token from CloudAccess and issues an OAuth token for the mobile device.
4. The Salesforce native app stores and uses an OAuth token to access Salesforce.com services. The token is reused for future sessions, so the user does not have to re-enter credentials as long as the token has not expired.

From the user's perspective, the process is as follows:

1. The user installs and sets up the mobile Salesforce app on a supported mobile device.
2. The user installs the NetIQ MobileAccess app on a supported mobile device.
3. The user opens the Salesforce mobile app on the device and enters the preconfigured custom domain.
4. The user is redirected to the NetIQ MobileAccess app and, if required, is asked for the PIN.
5. The user is redirected back to the Salesforce app and the standard OAuth use consent window appears.
6. The user selects **Allow** on the consent window and is SSO'd to the mobile Salesforce app. The user does not have to re-enter credentials until the OAuth session token expires.
7. When the user logs out of the mobile Salesforce app, the user sees the Salesforce.com login page.

NOTE: If the user does not have the MobileAccess app installed on the device (or if it was previously installed and then deleted), instead of being automatically authenticated, a NetIQ MobileAccess login screen appears in step 4 and the user must enter credentials.

Requirements for Mobile SSO to Salesforce

The following requirements must be met to enable SSO to Salesforce on mobile devices:

- ❒ When you complete the connector configuration at Salesforce to allow single sign-on for users, you must configure a custom domain at Salesforce and provide the custom domain URL to users.

  For more information, see Section 7.8.3, "Configuring the Connector for Salesforce," on page 109.

- ❒ When you configure the connector for Salesforce in CloudAccess, you may create an appmark specifically for single sign-on to the mobile Salesforce app, selecting **Native application** from the list of **Launch with** options.

  For more information, see Section 7.5, "Configuring Appmarks for Connectors," on page 91.

- ❒ SAML SSO works on mobile devices only if the NetIQ MobileAccess app is also installed and configured on the device. Without MobileAccess installed on mobile devices, only delegated authentication using SAML is available.

  For more information, see "Installing MobileAccess on a Mobile Device" in the *NetIQ® CloudAccess and MobileAccess Installation and Configuration Guide*.

7.8.7 Configuring Delegated Authentication in Salesforce

Salesforce allows two different types of authentication methods: SAML and delegated authentication. By default, Salesforce activates only the SAML authentication. SAML is available for browser-based authentication or for mobile devices. However, SAML SSO works on mobile devices only if the NetIQ MobileAccess app is also installed and configured on the device.

Delegated authentication must be activated on a per-Salesforce organization basis. This allows CloudAccess to support users authenticating with mobile devices as well as users authenticating with browsers.

The phone icon that CloudAccess displays on all the Salesforce connectors indicates that Delegated Authentication can be used with Salesforce. You must enable and configure Delegated Authentication in Salesforce, and enable it in the connector. For more information, see Step 2 and Step 8 in Section 7.8.3, “Configuring the Connector for Salesforce,” on page 109.

The following setup is required in CloudAccess in order for delegated authentication to work properly:

- The DNS name of the CloudAccess cluster must be publicly resolvable.
- The SSL certificate must be signed by a well-known certificate authority (CA).

To configure Salesforce for delegated authentication:

1. Follow the instructions in the Salesforce documentation to enable delegated authentication single sign-on for your organization.

   For more information, see Configuring Salesforce for Delegated Authentication (https://login.salesforce.com/help/doc/en/sso_delauthentication_configuring.htm).

2. After delegated authentication has been enabled at Salesforce, complete the following configuration steps:

   2a Log in to the Salesforce administration page.

   2b Click **Your Name** > **Setup** > **Security Controls** > **Single Sign-On Settings** > **Edit**.

   2c In the **Delegated Gateway URL** field, specify a value similar to the following: `https://`*`cloudaccess_public_dns_name`*`/osp/a/t1/auth/external/sfda`.

   2d Do not select **Force Delegated Authentication Callout**.

   This option affects the performance of user logins.

   2e Enable the **Is Single Sign-On Enabled** permission. Note that if you want to prompt users to validate their accounts, you must disable this option instead. For more information about the **Prompt Before Provisioning** option, see Section 7.4, “How CloudAccess Provisions User Accounts,” on page 86.

3. Configure a connector for Salesforce in CloudAccess as described in section Section 7.8.3, “Configuring the Connector for Salesforce,” on page 109, but deselect the **Delegated authentication single sign-on is disabled in Salesforce** option.

When end users authenticate to Salesforce through their mobile devices, they will authenticate entering identity source credentials, where the user name is specified in email format to match the user name in the Salesforce account.

For example, if Active Directory user `Ted` with password `password` has been provisioned to Salesforce domain `mydomain-dev-ed.my.salesforce.com`, the user name for login from a mobile device app such as Salesforce Chatter would be `Ted@mydomain-dev-ed.my.salesforce.com` and the password would be `password`.

7.8.8 Configuring the Salesforce Federation Identifier

The Salesforce connector uses the Federation attribute to store the user's Salesforce federation identity. You can use one of the following as the attribute type for all users:

- **GUID:** The federation identifier uses the adroitBISObjectID. Using a GUID is the default setting.
- **WorkforceID/employeeID:** If you select this option, the identity source must supply the value for the appliance and Salesforce connector to use. All of the current identity sources support a workforceID attribute:
  - workforceID (eDirectory)
  - employeeID (Active Directory)
  - workforceid (JDBC)

  If the user has a workforceID or employeeID in the identity source, the user's account is provisioned in Salesforce.

  If the user does not have a workforceID or employeeID in the identity source, the `connectors_SFORCE_XXXXX.log` file has a message that the provisioning activity for that user was vetoed. Add a workforceID/employeeID to the User object in the identity source. When the workforceID or employeeID is synchronized, the account is automatically provisioned.

When you configure the Salesforce connector, you can use the **Advanced Settings** > **Federation attribute** option to specify which attribute type to use. For more information, see Section 7.8.3, "Configuring the Connector for Salesforce," on page 109.

To change from using a GUID to using a workforceID/employeeID for the federation identity, or vice versa:

1. Log in with an appliance administrator account to the Admin page at

   `https://appliance_dns_name/appliance/index.html`

2. On the Policy Mapping page, de-provision users from Salesforce by removing the current policy mapping so that the users are marked as inactive in Salesforce.

   For information about policy mapping, see "Mapping Authorizations" in the *NetIQ® CloudAccess and MobileAccess Installation and Configuration Guide*.

3. On the Admin page, click the configured connector for Salesforce, then click **Configure**.
4. In the Salesforce connector configuration, click **Advanced Settings**, change the **Federation identifier** setting, then click **OK** and **Apply** to save and apply the change.
5. Redo the policy mapping to trigger re-provisioning of users to Salesforce. The federation identifier is modified to use the appropriate attribute.
6. (Conditional) If you changed the **Federal identifier** setting from GUID to workforceID/employeeID, verify that all users were provisioned.

   6a Check the `connectors_SFORCE_XXXXX.log` file for messages about any user objects that were not provisioned because they did not have a workforceID/employeeID.

   6b For each user who was not provisioned, add a workforceID/employeeID to the User object in the identity source.

   When the workforceID/employeeID is synchronized, the account is automatically provisioned.

   6c Repeat this process to ensure that all authorized users are provisioned.

7.9 Connector for NetIQ Access Manager

NetIQ has enhanced the integration between CloudAccess and NetIQ Access Manager 4.1.*x* to simplify the configuration. If you are using an older version of Access Manager you must use the generic connector for NetIQ Access Manager. For more information, see Using NetIQ® CloudAccess as a Trusted Identity Provider for NetIQ® Access Manager.

The connector for NetIQ Access Manager configures your Access Manager Identity Server as a service provider (SP) that consumes the authentication information from CloudAccess. This allows the connector for NetIQ Access Manager to provide federated single sign-on access to Access Manager through CloudAccess. It does not support provisioning. The connector allows CloudAccess to authenticate a user against an identity source and to share this authentication with Access Manager in order to establish the user's session. You can also configure CloudAccess to provide MobileAccess features for Access Manager users.

CloudAccess includes this connector with the appliance. The administration console displays the connector automatically once you have installed CloudAccess.

- Section 7.9.1, "Requirements for the Connector for Access Manager," on page 117
- Section 7.9.2, "Configuring the Connector," on page 117
- Section 7.9.3, "Configuring Appmarks for Protected Resources in Access Manager," on page 118

7.9.1 Requirements for the Connector for Access Manager

- ❒ A CloudAccess appliance, installed and configured. MobileAccess configuration is optional, depending on your user authentication needs.
- ❒ A NetIQ Access Manager 4.1.*x* system, installed and configured.

  Ensure that SSL communications are enabled for Identity Server and Access Gateway, and that both components are configured to trust the same signing certificate authority. For more information, see "Enabling SSL Communications" (https://www.netiq.com/documentation/netiqaccessmanager4/basicconfig/data/b6vcbhk.html) in the *NetIQ Access Manager Setup Guide* (https://www.netiq.com/documentation/netiqaccessmanager4/basicconfig/data/bookinfo.html).
- ❒ Access Manager user accounts for each user who wants the single sign-on service.
- ❒ If you use an eDirectory identity source for Access Manager and you need to provide access to Access Gateway protected resources that require a user name and password, you must enable Universal Password in eDirectory for the Access Manager LDAP connection.

  NOTE: Universal Password Retrieval options must be properly set in the configuration of the Universal Password policy in eDirectory, so that it allows the password to be retrieved from the Access Manager user store.

  For more information, see *Unable to retrieve Universal Password from eDirectory using PasswordFetchClass (TID 7007114)* (http://www.novell.com/support/kb/doc.php?id=7007114).

7.9.2 Configuring the Connector

You must configure the connector to work with your Access Manager system.

To configure the connector for Access Manager:

1. Log in with an appliance administrator account to the CloudAccess administration console at

   `https://appliance_dns_name/appliance/index.html`

2. Drag the connector for NetIQ Access Manager from the **Applications** palette to the **Applications** panel.
3. Specify the following information to configure the connector:
   - A display name
   - The IP address or DNS name of the Access Manager administration console server
   - An administrator user name and password for Access Manager
   - The clustered IP address of the Identity Server (IDP)
4. Click **Advanced**.
   - **4a** Select the attribute that contains a user's name identifier in the CloudAccess identity source.
   - **4b** Select the matching attribute in the Access Manager identity source.
5. Click the **Appmarks** tab, then create an appmark for one or more of the Access Manager protected resource URLs listed.

   For more information, see Section 7.5, "Configuring Appmarks for Connectors," on page 91.
6. Click **OK** to save the configuration.
7. On the Admin page, click **Apply** to commit the changes to the appliance.
8. Click **Policy** on the toolbar, then perform policy mapping to specify entitlements for identity source roles (groups).

As long as you have met the requirements, the CloudAccess appliance creates the required components in Access Manager to make it a service provider.

7.9.3 Configuring Appmarks for Protected Resources in Access Manager

After you have configured the connector for Access Manager, you can configure appmarks for protected resources in Access Manager.

The default appmark for the connector for Access Manager uses the **Destination URL** field from the configuration. If you did not specify the **Destination URL**, you will end up at the Access Manager home page for the default appmark.

For more information, see Section 7.5, "Configuring Appmarks for Connectors," on page 91.

7.10 Connector for Bookmarks

The connector for Bookmarks on the Applications palette enables you to create links to web applications that are accessible from the browser landing page or directly from the MobileAccess app on users' mobile devices.

You can also create links to other mobile applications from the MobileAccess app, though there is no single sign-on for these apps.

While there is no global list for these app URL schemes, you might find the following list helpful:

http://wiki.akosma.com/IPhone_URL_Schemes

7.10.1 Configuring the Connector for Bookmarks

The connector for Bookmarks is intended as a container for multiple appmarks for web applications. Configure the Bookmarks connector once, then configure as many appmarks as you need within the same Bookmarks connector so you do not clutter your Admin page.

To configure a bookmark connector:

1. Log in as an administrator to the CloudAccess administration console:

   `https://appliance_dns_name/appliance/index.html`

2. Drag and drop the **Bookmark** connector from the **Applications** palette to the **Applications** panel.

   The Configuration window opens automatically for the initial configuration. To view or reconfigure the settings later, click the connector icon, then click **Configure**.

3. Provide a display name for the bookmarked application. The display name appears on the Admin page of the administration console and in the MobileAccess app.
4. Click the **Appmarks** tab.
5. Click the plus (**+**) sign next to the default created appmark.
6. Rename the appmark to correspond to the bookmark URL.
7. (Conditional) Select the **Public** check box if you want the appmark to appear for all users, regardless of their entitlement to the application.
8. (Conditional) If you want users to be able to access the bookmarked application from their desktop browser landing page, select **Desktop browser** and complete the following steps:

   8a In the **URL** field, change the default value to the URL of the bookmarked application.

   8b Click **X** next to the default icon to delete it, then browse to and select a `.png` file to represent the application on the browser landing page.

9. (Conditional) If you want users to be able to access the bookmarked application from their mobile devices, select **iOS devices** or **Android devices** and specify the appropriate options as follows:

   9a From the **Launch with** list, select the viewer in which the application should appear on mobile devices: Safari, Chrome, or an internal viewer.

   NOTE: If the URL for the appmark points to a destination web server that uses a non-public signing certificate for SSL, configure the appmark to open in a Safari or Chrome browser. With an internal viewer, users will receive a certificate error and their mobile devices cannot display the page.

   9b Leave the **Launch URL** and **App installer URL** fields blank.

   9c In the **URL** field, type the same URL that you provided in step 7a or type a mobile-specific URL.

   9d Click **X** next to the default icon to delete it, then browse to and select a `.png` file to represent the application on the mobile device.

10 (Conditional) If you want to use the Bookmark connector to link to other mobile applications from the MobileAccess app, select **iOS devices** or **Android devices** and specify the following options:

   10a From the **Launch with** list, select **Native application**.

   10b In the **Launch URL** field, enter the mobile app URL scheme. For example, `fb://profile`.

   10c (Optional) In the **App installer URL** field, type the URL to install the application if it has not already been installed on the mobile device.

   10d Click **X** next to the default icon to delete it, then browse to and select a `.png` file to represent the bookmarked application.

11 Click **OK**.

12 On the Admin page, click **Apply** to commit the changes to the appliance.

13 Wait until the configuration changes have been applied on each node of the CloudAccess cluster.

To see the application on their mobile devices, users must perform a refresh using the standard “pull-to-refresh” action on the Applications page in the MobileAccess app. (This action is used in mail and other common applications on the mobile device.) How users access the bookmark appmark depends on how you configured the **Launch with** option.

7.11 Connector for OAuth2 Resources

The connector for OAuth 2 Resources provides simple authenticated access to a web service through CloudAccess. The connector allows CloudAccess to authenticate a user against your identity sources and to provide protected access to a destination web service.

The connector for OAuth2 Resources offers a simple authentication method as an alternative to federated single sign-on connectors that use SAML 2.0 or WS-Federation protocols. Protocols for federated access management provide a robust trust and security model that is an open standard and widely used. However, it does require the protocol’s code to be installed on the protected services. Consider using the connector for OAuth2 Resources for smaller services that do not require the full security and trust that SAML or WS-Federation provides, and just need a simple method to validate and get identity information from a trusted source (the CloudAccess identity provider in this case).

By implementing the open standard OAuth 2.0 protocol, the connector for OAuth2 Resources behaves as an OAuth2 Authorization Server and Resource Server using the Authorization Code flow as detailed in the OAuth 2.0 Authorization Framework document at http://tools.ietf.org/html/rfc6749#section-4.1 .

Using this connector, the CloudAccess appliance provides user authentication and all OAuth2 token creation and validation for access to a protected resource.

NOTE: The OAuth2 Resources connector provides SP-initiated authentication. It does not have an IDP-initiated mode.

Use the information in the following sections to configure a connector for OAuth2 Resources:

- Section 7.11.1, “Configuring the OAuth2 Client Application,” on page 121
- Section 7.11.2, “Configuring the Connector for OAuth2 Resources,” on page 121
- Section 7.11.3, “Supported OpenID Connect Schema,” on page 122

7.11.1 Configuring the OAuth2 Client Application

When you configure the connector for OAuth2 Resources on CloudAccess, the Client ID, Client Secret, and OAuth Endpoint URLs are created automatically. This information must then be used to configure the OAuth2 client application. All configuration activities at the OAuth2 client application are out of band.

Enforcement of authorization or access control beyond the initial authentication and token creation process is the responsibility of the OAuth client application, since the OAuth Resources connector does not currently support policy mapping in CloudAccess.

For information about configuring the OAuth client application, refer to your OAuth client application documentation.

7.11.2 Configuring the Connector for OAuth2 Resources

You can configure instances of the OAuth2 Resources connector in one of the following ways:

- An instance of the connector per OAuth client application. This is the simplest method conceptually and matches how SAML connectors are used.
- Multiple OAuth client applications all configured within a single instance of the OAuth2 Resources connector. This means that all OAuth2 client applications would use the same schema (OpenID Connect or native), and would use the same Client ID and Client Secret. This configuration is simple to configure and maintain, but care should be taken to include only clients of the same trust level in a connector instance. Because all clients share the same client ID and secret, if one of the clients is compromised in any way, they are all compromised. Any of them could also masquerade as another client in some cases.

(Optional) For each OAuth client application, you can manually create appmarks so the CloudAccess landing page shows an icon for connection to the OAuth2 client application. Appmarks should be configured to point to the URL of the OAuth2 client application that will start the OAuth2 authentication process.

To configure the connector for OAuth2 Resources:

1. Log in as an administrator to the CloudAccess administration console:

   `https://appliance_dns_name/appliance/index.html`

2. Drag and drop the **OAuth Resources** connector from the **Applications** palette to the **Applications** panel.

   The Configuration window opens automatically for the initial configuration. To view or reconfigure the settings later, click the connector icon, then click **Configure**.

3. On the **Configuration** tab, provide the following information:
   - **Display name**: Clearly identify the connector on the Admin page of the console.
   - **Schema**: Specify whether the attributes that CloudAccess sends to the OAuth client follow OpenID Connect standard naming or use the Native schema names defined internally on the appliance.
   - **Allowed OAuth Client URI(s)**: Specify the whole path or just the host name for the OAuth2 client application. Using only the host name allows all paths on that domain. Since OAuth2 depends on SSL as one of its core security mechanisms, HTTPS should always be specified. For more information about configuring redirect URIs, see the following document: http://tools.ietf.org/html/rfc6749#section-10.6.

- **OAuth Details (Client ID and Client Secret)**: Use this information to configure the OAuth2 client application.
- **OAuth Endpoints (Auth URL, Token URL, and Profile URL)**: Use this information to configure the OAuth2 client application.

4 Click the **Appmarks** tab, then review and edit the default settings for the appmark.

5 Click **OK** to save the configuration.

6 On the Admin page, click **Apply** to commit the changes to the appliance.

7 Wait until the configuration changes have been applied on each node of the CloudAccess cluster.

8 (Conditional) If Public access is disabled, click **Policy** in the toolbar, then perform policy mapping to specify entitlements for identity source roles (groups).

For more information, see "Mapping Authorizations" in the *NetIQ® CloudAccess and MobileAccess Installation and Configuration Guide*.

After the OAuth2 Resources connector and OAuth client application have been configured, end users can access the protected resource by browsing to the URL of the OAuth client application (by entering the URL directly into the browser, using a bookmark or the landing page appmark, and so forth). If the user is not already authenticated to the CloudAccess appliance, the browser is redirected to the CloudAccess login page and the user is prompted for login credentials. After a successful authentication or if the user is already authenticated to the appliance and is authorized to access the protected resource, the user gains access to the resource.

7.11.3 Supported OpenID Connect Schema

The OAuth Resources connector supports the OpenID Connect schema names listed in the following table.

Table 7-5 *OpenID Connect Schema*

Member	Type	Description
name	string	End user's full name in displayable form including all name parts, possibly including titles and suffixes, ordered according to the user's locale and preferences.
given_name	string	Given name(s) or first name(s) of the end user. Note that in some cultures, people can have multiple given names; all can be present, with the names being separated by space characters.
family_name	string	Surname(s) or last name(s) of the end user. Note that in some cultures, people can have multiple family names or no family name; all can be present, with the names being separated by space characters.
middle_name	string	Middle name(s) of the end user. Note that in some cultures, people can have multiple middle names; all can be present, with the names being separated by space characters. Also note that in some cultures, middle names are not used.

Member	Type	Description
preferred_username	string	Shorthand name that the end user wishes to be referred to at the RP, such as janedoe or j.doe. This value *may* be any valid JSON string including special characters such as @, /, or whitespace. This value *must not* be relied upon to be unique by the RP. (See Section 2.5.3 (http://openid.net/specs/openid-connect-basic-1_0-28.html#claim.stability) of the OpenID Connect Basic Client Profile 1.0 document.)
picture	string	URL of the end user's profile picture. This URL *must* refer to an image file (for example, a PNG, JPEG, or GIF image file), rather than to a Web page containing an image. Note that this URL *should* specifically reference a profile photo of the end user suitable for displaying when describing the end user, rather than an arbitrary photo taken by the end user.
email	string	end user's preferred email address. Its value *must* conform to the RFC 5322 (http://openid.net/specs/openid-connect-basic-1_0-28.html#RFC5322) addr-spec syntax. This value *must not* be relied upon to be unique by the RP, as discussed in Section 2.5.3 (http://openid.net/specs/openid-connect-basic-1_0-28.html#claim.stability) of the OpenID Connect Basic Client Profile 1.0 document.
gender	string	End user's gender. Values defined by this specification are female and male. Other values *may* be used when neither of the defined values is applicable.
birthdate	string	End user's birthday, represented as an ISO 8601:2004 (http://openid.net/specs/openid-connect-basic-1_0-28.html#ISO8601-2004)[ISO8601-2004] YYYY-MM-DD format. The year *may* be 0000, indicating that it is omitted. To represent only the year, YYYY format is allowed. Note that depending on the underlying platform's date related function, providing just year can result in varying month and day, so the implementers need to take this factor into account to correctly process the dates.
locale	string	End user’s locale, represented as a BCP47 (http://openid.net/specs/openid-connect-basic-1_0-28.html#RFC5646) [RFC5646] language tag. This is typically an ISO 639-1 Alpha-2 (http://openid.net/specs/openid-connect-basic-1_0-28.html#ISO3166-1) [ISO639 1] language code in lowercase and an ISO 3166-1 Alpha-2 (http://openid.net/specs/openid-connect-basic-1_0-28.html#ISO3166-1) [ISO3166 1] country code in uppercase, separated by a dash. For example, en-US or fr-CA. As a compatibility note, some implementations have used an underscore as the separator rather than a dash, for example, en_US; Implementations *may* choose to accept this locale syntax as well.

Member	Type	Description
phone_number	string	End user's preferred telephone number. E.164 (http://openid.net/specs/openid-connect-basic-1_0-28.html#E.164) [E.164] is *recommended* as the format of this Claim, for example, +1 (425) 555-1212 or +56 (2) 687 2400. If the phone number contains an extension, it is *recommended* that the extension be represented using the RFC 3966 (http://openid.net/specs/openid-connect-basic-1_0-28.html#RFC3966) [RFC3966] extension syntax, for example, +1 (604) 555-1234;ext=5678.

7.12 Connector for Simple Proxy

The connector for Simple Proxy provides reverse proxy access to your enterprise web service through CloudAccess. The connector allows CloudAccess to authenticate a user against your identity sources and to provide protected access to a destination web service. You can configure the connector to protect access to the document root of the web server, or to protect access only to a path within the document root of the web server.

Every web server is different. Two common simple proxy scenarios are:

- **Simple website:** If your web server provides a single web service, you can protect access to the entire site by creating a connector for Simple Proxy. The connector points to the document root of the web server.
- **Multiple-service website:** If your server provides multiple web services, you can create a separate connector for Simple Proxy for each destination web service. Each connector points to a different independent path within the document root of the web server.

If the web service requires user identity information to control access or content, you can configure the connector to inject the authenticated user's identity attributes in query strings and HTTP headers sent to the web service. Because the connector cannot send passwords, the connector cannot be used to provide single sign-on for web services that require passwords for access. It does not support provisioning.

CloudAccess includes this connector with the appliance. The connector is included automatically on the Applications palette of the Admin page. Each cluster supports multiple connectors for Simple Proxy.

Use the information in the following sections to configure a connector for Simple Proxy:

- Section 7.12.1, "Requirements for Simple Proxy," on page 124
- Section 7.12.2, "Viewing or Customizing the Attributes for Identity Injection," on page 125
- Section 7.12.3, "Configuring the Connector for Simple Proxy," on page 128

7.12.1 Requirements for Simple Proxy

The connector for Simple Proxy enables reverse proxy access to an enterprise web server behind your firewall. It can support web services that employ user identity information to control access or display if you enable the identity injection policies that insert an authenticated user's identity attributes in query strings or headers of requests it sends to the web server. For more information, see Section 7.12.2, "Viewing or Customizing the Attributes for Identity Injection," on page 125.

For each proxy web service, the web service's content should be self-contained in that path. If the service depends on files that reside in parallel paths on the web server, you can specify a path at a higher level in the document root's directory structure, or reorganize the site's contents as needed.

The connector for Simple Proxy does not support the following:

- **Protected resources that require a password:** This proxy solution cannot be used with protected web services or applications that require an LDAP password to be included in the identity injection. The appliance cannot send a user's password for a proxy application to the back end web service.

  If the web server needs the user's password, you must find a workaround. For example, you could specify a static string that is accepted for all users.
- **Site redirects:** This proxy solution does not support site redirects to locations outside the protected path. It cannot follow paths to alternate websites.

IMPORTANT: The Access Gateway for NetIQ Access Manager provides solutions for more complex reverse proxies that support password injection and redirects. For more information, see "Managing Reverse Proxies and Authentication" in the *NetIQ Access Manager Access Gateway Guide*.

Before you configure a connector for Simple Proxy, ensure that your setup meets the following requirements:

- ❒ A CloudAccess system, installed and configured.
- ❒ A web server, configured and running behind the corporate firewall. Ensure that you have configured the authentication procedures and identity injection policy for the web service.

  You need the following information:

  - The primary DNS name or IP address of the web server.
  - Alternative DNS names or IP addresses for the web server, if any.
  - The port number that the web server uses to listen for requests, such as 8080 (non-secure) or 8443 (secure SSL).
  - If the web server requires it, secure communications with HTTPS.

    If you use HTTPS, the value that you specify for the web server's DNS name or IP address in the connector must match the CN in the web server's SSL certificate.
- ❒ Determine which web services you need to protect for your web server, and which users require access to each one.

7.12.2 Viewing or Customizing the Attributes for Identity Injection

The connector for Simple Proxy can inject an authenticated user's identity attributes in query strings and headers of communications sent from the appliance to the destination web service. The web server might use this information to determine whether the user should have access to the resource.

It can also use the identity information to customize content on the web page. For example, when a user whose first name is Joe (as specified in the identity source) navigates to the destination web page, he might see “Welcome: Joe” at the top of his browser window.

- “Understanding Identity Attributes” on page 126
- “Viewing Identity Attribute Mappings to Identity Source Attributes” on page 127
- “Configuring Custom Identity Attributes” on page 128

Understanding Identity Attributes

In the connector for Simple Proxy, you can enable or disable the following identity injection policies. Both policies are enabled by default.

- **Inject Identity in Query:** If you enable this option, when a user navigates to the connector’s destination web service, the service receives all of the user’s identity attributes in the query string.

  WARNING: Injecting attributes in the query string could exceed the maximum URL length of 2083 characters.

- **Inject Identity in Header:** If you enable this option, when a user navigates to the connector’s destination web service, the service receives all of the user’s identity attributes as custom headers.

If you enable an injection policy, the connector sends all of the user’s identity attributes, even if the values are unavailable (empty). For some applications, this is still useful information and the web service can use it to make access or display decisions.

WARNING: If you use HTTP for communications between the connector and the web service, the injected identity attributes are available as clear text to network packet sniffers.

Although the proxy service runs behind the firewall, consider configuring the connector’s web service URL with HTTPS to protect the communication stream to the web service. If you use HTTPS, the value that you specify for the web server’s DNS name or IP address in the connector must match the CN in the web server's SSL certificate.

The attribute values in the query strings parameters or header parameters sent to the web server are based on the following options in the identity source user interface:

Identity Source Parameter	Query String Parameter	Header Parameter
ID	ID	X-ID
Email	Email	X-Email
User name	UserName	X-UserName
First name	FirstName	X-FirstName
Middle name	MiddleName	X-MiddleName
Last name	LastName	X-LastName
Full name	FullName	X-FullName
Preferred name	PreferredName	X-PreferredName
Generational qualifier	GenerationalQualifier	X-GenerationalQualifier
Gender	Gender	X-Gender
Phone	Phone	X-Phone
Birthdate	BirthDate	X-BirthDate
Street address	StreetAddress	X-StreetAddress
City	City	X-City
State	State	X-State
ZIP code	ZipCode	X-ZipCode
Country	Country	X-Country
Language	Language	X-Language
Identity Type	IdentityType	X-IdentityType
X-Custom1	XCustom1	X-XCustom1
X-Custom2	XCustom2	X-XCustom2
X-Custom3	XCustom3	X-XCustom3
X-Custom4	XCustom4	X-XCustom4
X-Custom5	XCustom5	X-XCustom5

The IdentityType parameter for query strings and headers indicates the type of identity source that the appliance uses to authenticate the user, such as Active Directory, eDirectory, Self-Service User Store (SSUS), and JDBC.

Viewing Identity Attribute Mappings to Identity Source Attributes

In your identity source, the identity attributes are mapped to identity source attributes. You can view mappings in your identity source connector. For example, in eDirectory, `ID` is mapped to the `guid` attribute, `User name` is mapped to the `cn`, and so on. You can change these mappings as needed for your environment, but any changes you make are global. You cannot change them on a per proxy or app basis.

To view identity attribute mappings in an identity source:

1. Log in as an administrator to the CloudAccess administration console:

   `https://`*`appliance_dns_name`*`/appliance/index.html`

2. In the **Identity Sources** panel, click the identity source, then click **Configure**.
3. Expand **Advanced Options**.
4. In the **Attribute Mappings** section, expand **Default** to view the list of the mappings of identity attributes to identity source attributes.

5 If you modify the settings, click **OK** to save your changes, and then click **Apply** on the Admin page.

   Do not continue until the changes are applied to all nodes of the appliance cluster.

6 Repeat this process for each identity source that manages users who will access the destination web server.

Configuring Custom Identity Attributes

An identity injection sends all identity attributes. You cannot specify only a subset of attributes, add attributes, or remove attributes. However, you can map the `X-Custom`<1-5> attributes to attributes in your identity source. Ensure that you map the appropriate identity source attribute to each custom attribute across all of the identity sources for users who will access the destination web server.

To configure custom identity attributes in your identity source:

1 Log in as an administrator to the CloudAccess administration console:

   `https://`*`appliance_dns_name`*`/appliance/index.html`

2 In the **Identity Sources** panel, click the identity source, then click **Configure**.

3 Expand **Advanced Options**, then use the **Attribute Mappings** section to map custom attributes (`X-Custom`<1-5>) to attributes in your identity source.

4 Click **OK** to save your changes, and then click **Apply** on the Admin page.

   Do not continue until the changes are applied to all nodes of the appliance cluster.

5 Repeat this setup for each identity source that manages users who will access the destination web server.

7.12.3 Configuring the Connector for Simple Proxy

Each connector for Simple Proxy can protect only a single web location. If the connector is set to protect the document root, then users can access all files served by the website. If the connector is set to protect a path under the document root, users can access only those files that reside in the path or its subdirectories.

The connector protects access to the web service based on its Appmarks settings. You can allow public access so that all users have access to the web service. You can alternatively deny public access and grant access to users of specific identity source roles by mapping authorization policies for them. Authorization policies dynamically control visibility of appmarks on the landing page for users after authentication. They enforce access when a user attempts to access a protected resource with an appmark, or when they directly browse the website.

To configure the connector for Simple Proxy:

1 Log in as an administrator to the CloudAccess administration console:

   `https://`*`appliance_dns_name`*`/appliance/index.html`

2 Drag and drop the connector for Simple Proxy from the **Applications** palette to the **Applications** panel.

   The Configuration window opens automatically for the initial configuration. To view or reconfigure the settings later, click the connector icon, then click **Configure**.

3 In the Configuration window on the **Configuration** tab, provide the following information:

Display name: Specify the display name for the reverse proxy service. This name is also the default name of the appmark that appears in the user interface.

Local path: Specify a unique path on the appliance that will be used in the URL to associate traffic for the remote web service, such as `/myservice` or `servicexyz`. The path will be appended to the DNS name of the cluster for accessing the resource, and will be removed from the request before forwarding it to the web server.

The local path must be unique across all connectors for Simple Proxy that you configure on the appliance. You can use alphanumeric characters a to z and 0 to 9, forward slashes (/), hyphens (-), and underscores (_). Spaces, uppercase characters, and other special characters are not supported. The path is not case sensitive. There is no length limit, but you should consider length restrictions for URLs and file system pathnames when you specify the character string.

Connects to: Specify the URL of the destination web service that you want to protect.

You can use HTTP (not secure) or HTTPS (secure) in the URL, depending on requirements of the web server. If you use HTTPS, the value that you specify for the DNS name or IP address must match the CN in the web server's SSL certificate. The connector automatically finds the SSL certificate and installs it for you if the URL uses HTTPS.

You can specify the IP address or DNS name of the web server. Specify the port number if it is needed to access the location.

Do one of the following:

- Specify the root of the web server in order to protect all resources in the document root of the web server, including its subdirectories and their content.

  For example, you can specify the URL in any of the following formats:

  ```
  http://10.20.30.40
  http://10.20.30.40:8080
  https://myweb.example.com
  https://myweb.example.com:8443
  ```

- Specify a path within the document root of the web server in order to protect only the resources in that path, including its subdirectories and their content. For example:

  Connects to: `http://151.155.160.14/examples`

  Appmark URL: `http://${PublicDNS}${PathFrag}`

  To specify a specific filename or to include query string data, add that information to the end of the URL configured in the appmark. For example:

  Connects to: `http://www.youtube.com/watch`

  Appmark URL: `http://${PublicDNS}${PathFrag}?v=jAZveG_ptVU&sns=em`

  Similarly, for a file name of `http://151.155.160.14/examples/servlets/helloworld.html`, specify the following:

  Connects to: `http://151.155.160.14/examples/servlets`

  Appmark URL: `http://${PublicDNS}${PathFrag}/helloworld.html`

Inject Identity in Query: Select this option to include the user identity attributes in the query strings that are sent to the **Connects to** URL. For more information, see Section 7.12.2, “Viewing or Customizing the Attributes for Identity Injection,” on page 125.

WARNING: Injecting attributes in the query string could exceed the maximum URL length of 2083 characters.

Inject Identity in Headers: Select this option to include the user identity attributes in the headers that are sent to the **Connects to** URL. For more information, see Section 7.12.2, "Viewing or Customizing the Attributes for Identity Injection," on page 125.

4 Expand **Advanced Options**, then configure the **Rewriter Options**:

The rewriter parses and searches the web content that passes through the appliance for URL references that qualify to be rewritten. URL references are rewritten when they meet the following conditions:

Strip Local path from query string: Enables URL references specified in the query strings to be rewritten with the published DNS name.

Strip Local path from POST data: Enables URL references specified in the post data to be rewritten with the published DNS name.

Strip Local path from REFERRER header: Enables URL references specified in the referrer headers to be rewritten with the published DNS name.

Alternative Host Names: URL references that match entries in this list are rewritten with the published DNS name. You can use any of the following formats. The entries are not case sensitive.

```
site.example.com
myhostname
10.10.2.10
http://<dns_name_or_ip_address>
http://<dns_name_or_ip_address>:port
https://<dns_name_or_ip_address>
https://<dns_name_or_ip_address>:port
```

You need to include names in this list if your web servers have the following configurations:

- If you have a cluster of web servers that are not sharing the same DNS name, you need to add their DNS names to this list.
- If your web server obtains content from another web server, the DNS name for this additional web server needs to be added to the list.
- If the web server listens on one port (for example, 80), and redirects the request to a secure port (for example, 443), the DNS name needs to be added to the list. This allows the response to be sent in the format that the user expects.
- If an application is written to use a private hostname, you need to add the private hostname to the list. For example, `http://<hostname>/index.html`.

5 Click the **Appmarks** tab, then review and edit the default settings for the appmark.

6 Click **OK** to save the configuration.

7 On the Admin page, click **Apply** to commit the changes to the appliance.

8 Wait until the configuration changes have been applied on each node of the CloudAccess cluster.

9 (Conditional) If Public access is disabled, click **Policy** in the toolbar, then perform policy mapping to specify entitlements for identity source roles (groups).

For more information, see "Mapping Authorizations" in the *NetIQ® CloudAccess and MobileAccess Installation and Configuration Guide*.

7.13 Creating Custom Connectors

CloudAccess provides the NetIQ Access Connector Toolkit (ACT) that allows you to create custom connectors. If you need help creating a custom connector to use with CloudAccess, Priority Support customers have the option to open a service request with NetIQ Technical Support (NTS) (http://

www.netiq.com/support). NTS is available to provide toolkit support as well as to configure the connectors to work with integrated applications. Additional information from the SaaS provider is usually required.

NOTE: Before you contact NetIQ Technical Support, please complete the appropriate worksheet for the connector type that you want to create. See "Custom Connector Worksheets".

The Access Connector Toolkit facilitates custom connector development efforts without coding or scripting. You can create custom connectors for identity-aware web service or applications that use the following authentication methods for single sign-on:

- SAML 2.0
- WS-Federation
- SAML 2.0 Inbound (SAML2 In)
- Basic SSO (forms-based)

After you create a connector, you must export it from the toolkit as a file that you can import into CloudAccess. You can use the CloudAccess administration console to import and enable the connector, and to create appmarks and to map policies for the web service or application.

- Section 7.13.1, "Accessing the Access Connector Toolkit," on page 131
- Section 7.13.2, "Toolkit Requirements," on page 132
- Section 7.13.3, "Federation Requirements for the Application Service Provider," on page 132
- Section 7.13.4, "Creating a SAML 2.0 Connector Template," on page 135
- Section 7.13.5, "Creating a WS-Federation Connector Template," on page 137
- Section 7.13.6, "Creating a SAML 2.0 Inbound (SAML2 In) Connector Template," on page 138
- Section 7.13.7, "Creating a Basic SSO Connector Template," on page 142
- Section 7.13.8, "Modifying a Connector," on page 144
- Section 7.13.9, "Exporting a Connector Template," on page 144
- Section 7.13.10, "Importing and Configuring Custom Connectors," on page 145

7.13.1 Accessing the Access Connector Toolkit

The Access Connector Toolkit is a web application that you access through the CloudAccess appliance. Log in to the toolkit using CloudAccess administrator credentials at:

`https://appliance_dns_name/css/toolkit`

The Access Connector Toolkit does not currently provide a logout option, though the session does time out after 60 minutes of inactivity. Ensure that you close the browser after you finish working in the Access Connector Toolkit.

7.13.2 Toolkit Requirements

The Access Connector Toolkit is a web application that ships with CloudAccess. You can use the Access Connector Toolkit to create custom connectors if you have a CloudAccess license as well as appropriate accounts with the destination services.

- "License Information" on page 132
- "Toolkit Compatibility" on page 132
- "Provisioning Support" on page 132

License Information

The Access Connector Toolkit is a CloudAccess-only feature. If you purchased MobileAccess without CloudAccess, your license entitles you to a 90-day trial of CloudAccess, including the Access Connector Toolkit. At the end of the trial period, you are expected to purchase the appropriate license for CloudAccess or discontinue use of the CloudAccess-only features. For more information, see "Understanding Product Licensing" in the *NetIQ® CloudAccess and MobileAccess Installation and Configuration Guide*.

Toolkit Compatibility

The Access Connector Toolkit contains new functionality for the CloudAccess 2.1 release. In order to update an existing custom connector template with the new functions, you can import the template into the new toolkit, and then export the template again. The updated connector template contains the new functionality.

Templates that you create with the new toolkit are not backwards compatible with prior releases of the toolkit. You cannot import a connector from CloudAccess 2.1 into a toolkit that came with a prior version of CloudAccess. The import fails.

Provisioning Support

Provisioning is supported only through connectors created by NetIQ. At this time, you cannot create a custom connector template that supports provisioning user accounts to the connected system.

Account provisioning is not supported for users in a SAML 2.0 Inbound (SAML2 In) unmanaged internal identity store or in a Self-Service User Store (SSUS) identity source. For more information, see Section 7.4.1, "Requirements for Provisioning," on page 86.

7.13.3 Federation Requirements for the Application Service Provider

As you explore the features of the Access Connector Toolkit, refer to the definitions in this section to understand the type of information you will need to collect from the destination web service or application.

assertion

A SAML 2.0 assertion is a package of identity attributes for an authenticated user that is sent from the trusted identity provider to the service provider.

assertion properties

The properties of the assertion include the following information:

- The recipient of the assertion.
- The LDAP identity attribute to use when federating users with the destination application service provider. Does the NameID require an email address format, or does it require unspecified format?
- The URL where CloudAccess should redirect the end user's session after the user logs in successfully with the URL provided on the connector configuration page.
- The binding method to use for identity information sent to the destination provider. For SAML 2.0, the only supported binding method is POST.

assertion attributes

The provider should provide a technical document that describes the attributes that are required for an assertion, such as the user's name or email address. It can include the attributes that are required to assign roles. The SAML assertion typically requires the nameID attribute. You must map the SAML assertion attributes to the matching attributes in your identity source.

entityID

The entityID is a field from the metadata that uniquely identifies that particular service provider, such as *`sp_domain_name`*.

For example:

google.com

The entity ID might use information from the federation instructions, or from a setting completed on the Configuration page when you deploy the connector.

federation instructions

The federation instructions provide the information that you will use to configure federation for CloudAccess on the service provider site. The information identifies where on the service provider's site to find the federation configuration capability as well as the field values and other guidance that you need to complete the required information.

When you configure the connector, the federation instructions will automatically provide the following information about your appliance as the identity provider:

- The URL for single sign-on

  `https://`*`appliance_dns_name`*`/osp/a/t1/auth/saml2/sso`

- The URL for single logout

  `https://`*`appliance_dns_name`*`/osp/a/t1/auth/app/logout`

- The URL for the identity provider's entityID

  `https://`*`appliance_dns_name`*`/osp/a/t1/auth/saml2/metadata`

- The X.509 signing certificate for the appliance

  The web service or application uses the certificate to set the trust relationship with CloudAccess.

NOTE: When you copy the appliance's signing certificate, ensure that you include all leading and trailing hyphens in the certificate's Begin and End tags.

It provides the following information about your appliance if the login is initiated by the service-provider, such for connectors that use the WS-Federation protocol:

- The WS-Federation Passive URL
- The X.509 signing certificate for the appliance

metadata

The metadata is the configuration information that the application service provider uses to establish communications with the identity provider in an federation trust relationship. This usually includes a login URL or a customer-specific domain name, which is called the Assertion Consumer Service URL. Service providers allow you to export the required metadata to an XML file, or they provide the metadata in a public URL. The auto-generated metadata file from the service provider will not work as is. You must manually change the values to match your actual deployment environment.

The metadata usually includes the following information:

- The entityID for the service provider.
- The URL that receives the user identity information.
  - For SAML 2.0, the Assertion Consumer Service URL is where the assertion is posted by the browser. For example:

    `https://www.google.com/a/${customer-domain}/acs`
  - For WS-Federation, the Login URL is where the security token is posted by the browser. It corresponds to the `PassiveRequestorEndpoint` field from the metadata.
  - For SAML2 In, the Single Sign-on Service URL is where the AuthnRequest will be posted. It corresponds to the `SingleSignOnService` field with a Post binding from the metadata.

    `https://accessmanager.base.url/nidp/saml2/sso`
- The logout URL corresponds to the `SingleLogoutService` field from the metadata.
- The logout URL Binding (HTTP Post or Redirect)

  The logout response URL
- The X.509 signing certificate

protocol binding

The protocol binding is the method used for transmitting assertions between the authenticating identity provider and the service provider. CloudAccess supports the Redirect and Post bindings for service-provider-initiated SSO, and the Post binding for identity-provider-initiated SSO.

nameID

The nameID is the attribute in the identity source that uniquely identifies the user. You must know whether this attribute requires the email address format or an unspecified format.

new settings

The new settings are appliance-specific settings that you want to allow the administrators to set when they configure the connector for an appliance.

For example:

- Customer-specific sections of the Assertion Consumer Service URL
- Connector-specific setting, such as a customer domain

security token

A WS-Federation security token is a package of identity attributes for an authenticated user that is sent from the trusted identity provider to the service provider. The provider should provide a technical document that describes the attributes that are required for the token, such as the user's name or email address. It can include the attributes that are required to assign roles.

signing certificate

The signing certificate is the X.509 certificate that identifies CloudAccess to the service provider. If you specify that the certificate is required by the service provider, the template automatically retrieves the appliance's certificate and inserts it in the Federation Instructions when you deploy the connector. You use the certificate when you set up the federated single sign-on for the application.

template properties

The template properties define the following information for the connector:

- Type of connector and type name (based on the template wizard)
- The unique name for the template file (target name)
- A brief description used as the connector name
- A 3-digit version number (ex: 1.0.0)
- A custom graphic to use for the icon that represents the connector on the Admin page.

7.13.4 Creating a SAML 2.0 Connector Template

To create a connector for single sign-on with SAML 2.0, you can use the **SAML2** option in the Access Connector Toolkit.

- "SAML 2.0 Requirements for the Application Service Provider" on page 135
- "Planning for a SAML 2.0 Connector" on page 136
- "Creating a SAML 2.0 Connector Template for an Application" on page 136

SAML 2.0 Requirements for the Application Service Provider

To create a custom SAML 2.0 connector for a destination application, ensure that the service provider meets the following protocol-specific requirements:

❒ Supports identity federation using the SAML 2.0 protocol.

For more information about SAML, see the OASIS website (https://wiki.oasis-open.org/security/FrontPage).

❒ Supports the SAML web browser single sign-on profile, with the Redirect and POST bindings for service-provider-initiated SSO, and the POST binding for identity-provider-initiated SSO.

❒ Provides a capability in the application's administration console that allows you to enable and configure SAML SSO with CloudAccess as the identity provider.

❒ Provides technical documents that describe the application's SAML federation requirements, metadata, and assertions.

Planning for a SAML 2.0 Connector

Before you attempt to create the SAML 2.0 connector, you must collect information about the destination web service or application. For more information, see Section 7.13.3, "Federation Requirements for the Application Service Provider," on page 132.

Ask the application service provider the following types of questions to gather the required information:

- What does your SAML assertion look like?
- Do you have a SAML metadata document? What fields, if any, are customer-specific?
- Does your service support the SAML single logout protocol?
- What are the required configuration steps in your application to set up federation?
- What information do you provide to customers when they are setting up federation with their identity source?

NOTE: You can use a worksheet to organize the information. See "Worksheet for SAML or WS-Federation Custom Connectors".

Creating a SAML 2.0 Connector Template for an Application

A SAML 2.0 connector template consists of multiple components for federation, metadata, and assertion information.

To create a custom SAML 2.0 connector:

1. Log in as an administrator to the Access Connector Toolkit.
2. Click **New** > **SAML2**.

   The connector **Type** is `SAML2`. The **Type Name** is `Generic SAML2 Connector`.
3. On the **Template** tab, complete the following information:
   - Template properties
   - Whether the service provider requires a signing certificate
   - Federation instructions for the service provider
   - New settings that need to be collected on the Configuration page of the connector
4. Click the **Metadata** tab, then use one of the following methods to specify the metadata:
   - Select Request, the specify the source URL to retrieve the metadata.
   - Complete the fields to manually generate the metadata.
   - Import the values from a file or URL, and modify them for your deployment environment.
5. Click the **Assertion** tab, then define the properties and attributes required for the assertion.

   5a On the **Properties** subtab, specify the properties for the assertion.

   5b On the **Attributes** subtab, click **New**, specify and define the identity attribute, then click **Save**.

   5c (Conditional) If the service provider requires other identity attributes for an assertion, repeat Step 5b to map the SAML assertion attribute to an attribute in your identity source.
6. (Optional) If it is supported, create the provisioning definitions. For more information, see "Provisioning Support" on page 132.

7 Click **Save** to save the new connector template.

8 Proceed to Section 7.13.9, “Exporting a Connector Template,” on page 144 to finish creating the new connector.

7.13.5 Creating a WS-Federation Connector Template

To create a connector for single sign-on with WS-Federation, you can use the **WS-Fed** option in the Access Connector Toolkit.

- “WS-Federation Requirements for the Application Service Provider” on page 137
- “Planning for a WS-Federation Connector” on page 137
- “Creating a WS-Federation Connector Template for an Application” on page 138

WS-Federation Requirements for the Application Service Provider

To create a custom WS-Federation connector for a destination application, ensure that the service provider meets the following protocol-specific requirements:

❒ Supports identity federation using the WS-Federation protocol.

For more information about WS-Federation, see the OASIS website (http://docs.oasis-open.org/wsfed/federation/v1.2/os/ws-federation-1.2-spec-os.html) or see the MSDN Library article (http://msdn.microsoft.com/en-us/library/bb498017.aspx).

❒ Supports the WS-Federation Passive Requestor Profile.

❒ Provides a capability in the application’s administration console that allows the customer to enable and configure WS-Federation SSO.

❒ Provides technical documents that describe the application’s WS-Federation federation requirements, metadata, and security tokens.

Planning for a WS-Federation Connector

Before you attempt to create a WS-Federation connector, you must collect information about the destination web service or application. For more information, see Section 7.13.3, “Federation Requirements for the Application Service Provider,” on page 132.

Ask the web service or application vendors the following types of questions to gather the require information:

- What does your WS-Federation security token look like?
- Do you have a WS-Federation metadata document? What fields, if any, are customer-specific?
- What are the required configuration steps in your application to set up federation?
- What is the information that you provide to customers when they are setting up federation with their identity source?

NOTE: You can use a worksheet to organize the information. See “Worksheet for SAML or WS-Federation Custom Connectors”.

Creating a WS-Federation Connector Template for an Application

A WS-Federation connector template consists of multiple components for federation, metadata, and assertion information.

To create a custom connector:

1. Log in as an administrator to the Access Connector Toolkit.
2. Click **New** > **WSFed**.

   The connector **Type** is `WS-Fed`. The **Type Name** is `Generic WS-Fed Connector`.
3. On the **Template** tab, complete the following information:
   - Template properties
   - Whether the service provider requires a signing certificate
   - Federation instructions for the service provider
   - New settings that need to be collected on the Configuration page of the connector
4. Click the **Metadata** tab, then use one of the following methods to specify the metadata:
   - Select Request, the specify the source URL to retrieve the metadata.
   - Complete the fields to manually generate the metadata.
   - Import the values from a file or URL, and modify them for your deployment environment.
5. Click the **Assertion** tab, then define the properties and attributes required for the security token.

   5a On the **Properties** subtab, specify the properties for the assertion.

   5b On the **Attributes** subtab, click **Predefined**, click the identity attribute, modify the definition if needed, then click **Save**.

   If a predefined option does not exist, use **New** to define it.

   5c (Conditional) If the service provider requires other identity attributes for an assertion, repeat Step 5b to map the WS-Federation attribute to an attribute in your identity source.
6. (Optional) Create the provisioning definitions. For more information, see "Provisioning Support" on page 132.
7. Click **Save** to save the new connector template.
8. Proceed to Section 7.13.9, "Exporting a Connector Template," on page 144 to finish creating the new connector.

7.13.6 Creating a SAML 2.0 Inbound (SAML2 In) Connector Template

To allow the appliance to be a SAML 2.0 service provider, you can create a SAML 2.0 Inbound connector using the Access Connector Toolkit. SAML2 Inbound as an identity source is not available during initialization of the appliance. However, after you export the connector and import it in the appliance, the SAML2 In connector appears as an identity source. You configure an instance of the identity source with information about an appropriate identity provider in order to enable the service provider functionality of the appliance, and to allow the identity provider to send a SAML token to the appliance using the SAML 2.0 POST profile.

After you configure the SAML2 In identity source, the appliance login page provides a link to the login page of the SAML 2.0 identity provider, located to the left of the user name and password login options. The SAML 2.0 users log in through the identity provider to gain access to the appliance landing page.

- "Understanding SAML2 In Identity Sources" on page 139
- "Requirements for Using SAML2 In Identity Sources" on page 140
- "SAML2 In Requirements for the Identity Provider" on page 141
- "Planning for a SAML2 In Connector" on page 141
- "Creating a SAML2 In Connector for an Identity Provider" on page 142

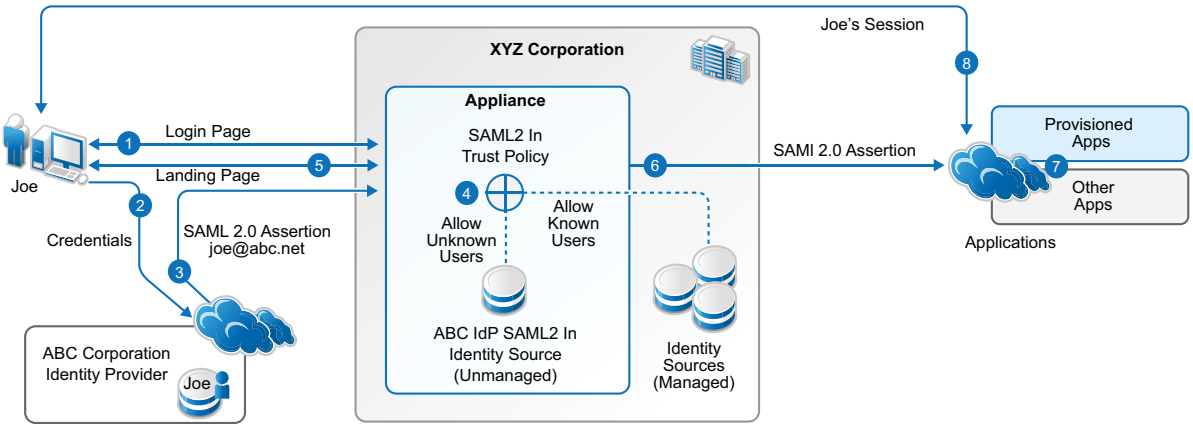
Understanding SAML2 In Identity Sources

A SAML2 In identity source can trust assertions from users who log in through the SAML2 In identity provider as known users or unknown users.

When you configure the SAML2 In identity source to accept assertions only for known users, the identity source accepts the authentication for users from the SAML2 In identity provider who have a matching email address in any one of the managed identity sources. The identity source denies access to the unmatched users. The known users can access applications according to the authorizations you map for their roles (groups) in the managed identity sources. Account provisioning and application access works the same as when the user logs in with corporate credentials through the appliance.

When you configure the SAML2 In identity source to accept assertions only for unknown users, the identity source accepts the authentication for all users from the SAML2 In identity provider, which should be a unique set of users as defined by email addresses. The identity source creates unique user objects in an unmanaged internal identity store for the identity provider. The unknown users can access applications according to the authorizations you map for their roles (groups) in the SAML2 In identity source.

Figure 7-4 *Using the Appliance as a Service Provider with the SAML2 In Identity Source*



The following is the experience for the SAML 2 In user:

1. The user goes to the appliance login page, then clicks the link for the identity provider.
2. The user receives the login page from the identity provider, and sends credentials.

3. The identity provider authenticates the user, then sends a SAML 2.0 assertion via the user's browser, to the appliance using the SAML 2.0 POST profile. The user identity is based on an email address.
4. The SAML 2 In identity source applies the trust policy that you configured for the identity provider:
   a. **Allow access for unknown users:** The appliance accepts authentication for all users from the identity provider, and creates a unique user object for the user in an internal identity store, based on the email address in the assertion.
   b. **Allow access for known users:** The appliance accepts authentication for users who have a matching identity in any of the managed identity sources, based on the email address in the assertion.
5. The appliance sends the landing page to the authenticated user.
6. When the user selects an appmark, the appliance builds an assertion for the user identity based on the SAML2 In trust policy:
   - **Allow access for unknown users:** The user's identity attributes are based on the user's information in the SAML2 In unmanaged internal identity store for the identity provider.
   - **Allow access for known users:** The user's identity attributes are based on the user's information in one of the managed identity sources.
7. The user accesses applications based on the entitlements that are associated with their logged-in identity.
8. The application service provider establishes a session with the user.

Requirements for Using SAML2 In Identity Sources

Consider the following requirements when you configure a SAML2 In identity source to allow access only for unknown users:

- ❒ The users who authenticate through the SAML2 In identity provider have no identities in the appliance's managed identity sources. That is, the users in the internal identity store have a unique identity for the appliance based on their email addresses.
- ❒ If a user has an identity in any of the appliance's managed identity sources, while the user is logged in through their identity provider account, the user cannot access applications based on the entitlements associated with the managed user account.
- ❒ Account provisioning is not supported for the users in the SAML2 In unmanaged internal identity store. Because these users do not have a workforceID, they cannot be provisioned for or access the SaaS applications that depend on the workforceID attribute for authentication, such as Google Apps and Salesforce.

  For more information, see Section 7.4.1, "Requirements for Provisioning," on page 86.
- ❒ Users in a SAML2 In internal identity store are not supported in administration roles for the appliance.

SAML2 In Requirements for the Identity Provider

To create a custom SAML 2.0 Inbound connector for an identity provider, ensure that the identity provider meets the following requirements:

- ❒ Supports identity federation using the SAML 2.0 protocol.

  For more information about SAML, see the OASIS website (https://wiki.oasis-open.org/security/FrontPage).

- ❒ Supports the SAML web browser single sign-on profile, with the Redirect and POST bindings for service-provider-initiated SSO, and the POST binding for identity-provider-initiated SSO.

- ❒ Provides a capability in the application's administration console that allows the customer to enable and configure SAML SSO.

  When you configure the SAML2 In connector, the **Federation Instructions** provide the information that you will need to set up the federation for CloudAccess in the identity provider. This information includes the metadata, a signing certificate for the appliance, the field values to use, and other guidance.

  The SAML 2.0 metadata for the appliance does not contain SAML 2.0 service provider information by default. You must configure at least one instance of a SAML2 In identity source before the appliance publishes service provider information in its metadata. To verify that a SAML2 In identity source is properly configured, open the appliance's metadata and search for the `SPSSODescriptor` tag.

  For information about importing and configuring the SAML2 In connector, see Section 7.13.10, "Importing and Configuring Custom Connectors," on page 145.

- ❒ Provides technical documents that describe SAML federation requirements, metadata, and assertions.

- ❒ Provides an Email attribute for every user. You will map this attribute to the SAML2 In NameID attribute.

  The SAML2 In identity source uses the value mapped to the NameID attribute in an assertion in order to uniquely identify the user. For more information about the role of email addresses for SAML2 In users, see "Requirements for Using SAML2 In Identity Sources" on page 140.

Planning for a SAML2 In Connector

Before you attempt to create the connector, you must collect information about the originating identity provider. Ask the identity provider vendors the following types of questions to gather the required information:

- What does your SAML assertion look like?
- Do you have a SAML metadata document? What fields, if any, are customer-specific?
- Does your service support the SAML single logout protocol?
- What are the required configuration steps in your application to set up federation?
- What is the information that you provide to customers when they are setting up federation?

NOTE: You can use a worksheet to organize the information. See "Worksheet for SAML In Custom Connectors".

Creating a SAML2 In Connector for an Identity Provider

A SAML2 In connector template consists of multiple components for federation, metadata, and assertion information.

To create a custom connector template:

1. Log in as an administrator to the Access Connector Toolkit.
2. Click **New** > **SAML2 In**.

   The connector **Type** is `SAML2 In`. The **Type Name** is `Generic SAML2 In Connector`.
3. On the **Template** tab, complete the following information:
   - Template properties
   - Whether the service provider requires a signing certificate
   - Federation instructions for the service provider
   - New settings that need to be collected on the Configuration page of the connector
4. Click the **Metadata** tab, then use one of the following methods to specify the metadata:
   - Select **Request**, then specify the source URL to retrieve the metadata.
   - Complete the fields to manually generate the metadata.
   - Import the values from a file or URL, and modify them for your deployment environment.
5. Click the **Assertion** tab, then define the properties and attributes required for the security token.

   5a On the **Properties** subtab, specify the properties for the assertion.

   5b On the **Attributes** subtab, click **Predefined**, click the identity attribute, modify the definition if needed, then click **Save**.

   Set **NameID** to the identity provider attribute that contains the user's email address.

   If a predefined option does not exist, use **New** to define it.

   5c (Conditional) If the service provider requires other identity attributes for an assertion, repeat Step 5b to map the WS-Federation attribute to an attribute in your identity source.
6. Click **Save** to save the new connector template.
7. Proceed to Section 7.13.9, "Exporting a Connector Template," on page 144 to finish creating the new connector.

7.13.7 Creating a Basic SSO Connector Template

A connector for basic single sign-on uses HTML Forms to populate the authentication information. To create a custom connector for Basic SSO, you must define the HTML form for the desired application.

- "Basic SSO Requirements" on page 142
- "Planning for Basic SSO" on page 143
- "Creating a Basic SSO Connector Template for a Web Service" on page 143

Basic SSO Requirements

Note the following requirements before you create a custom connector for Basic SSO:

❒ The application or web service must support HTML Forms.

For more information, see www.w3.org (http://www.w3.org/TR/html401/interact/forms.html).

- ❒ The connector supports user access to destination websites only through Chrome and Firefox web browsers running on a desktop or laptop computer. The connector works with the Basic SSO extension to securely collect, store, retrieve, and replay users' credentials for their destination websites.

  The MobileAccess app supports the secure retrieval and replay of previously stored credentials for websites that users access through the landing page on supported mobile devices.

- ❒ A user must install the Basic SSO extension in a supported browser one time on each desktop or laptop they use to access the Basic SSO websites.

  For Chrome, the extension is available for free from the Google Play Store. If it is not installed when the user accesses the application through CloudAccess, CloudAccess prompts the user to go to the Google Play Store and install it. The extension is added to the Chrome Extensions list, with the following permissions:

  - Access your data on all websites
  - Access your tabs and browsing activity

  For Firefox, the extension is available through Add-ons. The Firefox extension behaves the same way as the Chrome extension.

Planning for Basic SSO

Before you attempt to create the connector, you must collect information about the format of the HTML form on the login page of the web service or application. For example:

- What is the domain URL for the web service or application?
- What is the login page for the web service or application?
- What is the form ID or name for the user name?
- What is the form ID or name for the user password?
- What input type is used for the form (button, image, string)?

NOTE: You can use a worksheet to organize the information. See "Worksheet for Basic SSO Custom Connectors".

Creating a Basic SSO Connector Template for a Web Service

A Basic SSO connector template consists of multiple components. CloudAccess contains an interface that allows you to create the components in one place.

To create a connector template for Basic SSO:

1. Log in as a CloudAccess administrator to the Access Connector Toolkit at:

   `https://appliance_dns_name/css/toolkit`

2. Click **New** > **Basic SSO**.

   The connector **Type** is `Basic SSO`. The **Type Name** is `Generic Basic SSO Connector`.

3. On the **Template** tab, complete the template properties:

   - The unique name for the template file (target name). This name cannot include spaces or special characters.
   - A brief description of the connector.

- A 3-digit version number (ex: 1.0.0).
- A custom graphic to use for the icon that represents the connector on the Admin page.

4 (Conditional) If you need to specify a variable for the destination URL, under **Settings**, click **New**, then define the variable settings.

For example, some websites require an assigned keyword or organization name for the URL assigned to your company's account.

5 Click the **Forms** tab and create the form for the Basic SSO connector.

The **Forms** tab allows you to define the HTML form for the appropriate application and platform. You define the HTML form fields that are required to populate the form correctly. Use the information from w3.org (http://www.w3.org/TR/html401/interact/forms.html) to create the form.

Sometimes the application's HTML form is different for the desktop application or the mobile application. This means you must create multiple forms for the application. You can use the same fields for all three platforms or define a unique form for each platform.

6 Click **Save** to save the new connector template.

7 Proceed to Section 7.13.9, "Exporting a Connector Template," on page 144 to finish creating the new connector.

7.13.8 Modifying a Connector

You can modify the definition information for a connector by importing it in the Access Connector Toolkit. For example, you can import an existing connector to update its definition to the latest features available for connectors.

1 Obtain a copy of the connector's ZIP file.

2 Log in as a CloudAccess administrator to the Access Connector Toolkit at:

```
https://appliance_dns_name/css/toolkit
```

3 Click **Import**, browse to select the connector's ZIP file, then click **OK**.

The connector appears in the list of connector templates.

4 Click the **Edit** icon next to the **Display Name** for the connector template to open it in the Edit Connector Template window.

5 Modify the connector template settings as desired.

6 Click **Save** to apply the changes.

7 Click the **Export** icon next to the **Display Name** for the connector template.

8 Save the ZIP file for use on this or another CloudAccess system.

9 Proceed to Section 7.13.10, "Importing and Configuring Custom Connectors," on page 145.

7.13.9 Exporting a Connector Template

After you create a connector template, you must use the Access Connector Toolkit to export it in a compressed ZIP file that you can import to any CloudAccess system. You then import the connector template in the CloudAccess administration console to make it available in the **Applications** palette.

To export the connector template:

1 Log in as a CloudAccess administrator to the Access Connector Toolkit at:

```
https://appliance_dns_name/css/toolkit
```

2 Click the **Export** icon next to the **Display Name** for the connector template.

3 Save the ZIP file for use on this or another CloudAccess system.

4 Proceed to Section 7.13.10, “Importing and Configuring Custom Connectors,” on page 145.

7.13.10 Importing and Configuring Custom Connectors

CloudAccess allows you to import and configure custom connectors that you create with the Access Connector Toolkit, or that are created for you by NetIQ Technical Support or NetIQ partners.

After you export a custom connector, you must import its ZIP file to CloudAccess to make it available in the **Applications** palette of the administration console. Thereafter, you can enable and manage the connector as you do the connectors for applications that shipped with the appliance. The custom connector might require additional configuration, depending on the single sign-on method you use.

The destination application might also require additional configuration, depending on the application and the federation method. The destination applications for connectors for Basic SSO do not require additional configuration.

- “SAML2 and WS-Fed Custom Connectors” on page 145
- “SAML2 In Custom Connectors” on page 146
- “Basic SSO Custom Connectors” on page 147

SAML2 and WS-Fed Custom Connectors

To import and configure a custom connector for SAML2 and WS-Federation:

1 Log in as an administrator to the CloudAccess administration console:

`https://appliance_dns_name/appliance/index.html`

2 Import the custom connector to the **Applications** palette.

2a Copy the custom connector ZIP file to the computer where you administer CloudAccess.

2b On the Admin page, click the **Tools** icon on the toolbar, then click **Import connector template**.

2c Browse to and select the custom connector ZIP file, then click **Import**.

The connector appears in the **Applications** palette.

3 Drag the new custom connector from the **Applications** palette to the **Applications** panel.

The Configuration window opens automatically for the initial configuration. To view or reconfigure the settings later, click the connector icon, then click **Configure**.

4 Complete the connector settings on the **Configuration** tab.

The steps to configure the connector are determined by the information you added to the connector template.

5 Expand the **Federation Instructions**, then copy and paste the instructions into a text file to use when you configure the destination application.

NOTE: You must use a text editor that does not introduce hard returns or additional white space. For example, use Notepad instead of Wordpad.

---

6 Click the **Appmarks** tab, then review and edit the default settings for the appmark.

For more information, see Section 7.5, “Configuring Appmarks for Connectors,” on page 91.

7 Click **OK** to save the configuration.

8 On the Admin page, click **Apply** to commit the changes to the appliance.

9 Wait until the configuration changes have been applied on each node of the CloudAccess cluster.

10 Log in to the service provider as the account administrator, then configure the federation for CloudAccess in the application's administration console.

Use the information from the **Federation Instructions** in Step 5 to complete the setup.

NOTE: When you copy the appliance's signing certificate, ensure that you include all leading and trailing hyphens in the certificate's Begin and End tags.

---

11 In the CloudAccess administration console, click **Policy** on the toolbar, then perform policy mapping to specify entitlements for the SAML 2.0 Inbound users to the service provider application.

For more information, see "Mapping Authorizations" in the *NetIQ® CloudAccess and MobileAccess Installation and Configuration Guide*.

12 After you complete the configuration, users can log in through CloudAccess to single sign-on to the service provider's system. The CloudAccess login page URL is:

`https://appliance_dns_name`

SAML2 In Custom Connectors

Before you begin, ensure that you understand the trust policy settings for the SAML2 In identity sources. For more information, see "Understanding SAML2 In Identity Sources" on page 139 and "Requirements for Using SAML2 In Identity Sources" on page 140.

To import and configure a custom connector for SAML2 In as an identity source:

1 Log in as an administrator to the CloudAccess administration console:

`https://appliance_dns_name/appliance/index.html`

2 Import the custom SAML2 In connector to the **Applications** palette.

2a Copy the custom connector ZIP file to the computer where you administer CloudAccess.

2b On the Admin page, click the **Tools** icon on the toolbar, then click **Import connector template**.

2c Browse to and select the custom connector ZIP file, then click **Import**.

The connector appears in the **Identity Sources** palette.

3 Drag the new custom connector from the **Identity Sources** palette to the **Identity Sources** panel.

The Configuration window opens automatically for the initial configuration. To view or reconfigure the settings later, click the connector icon, then click **Configure**.

4 Complete the connector settings on the **Configuration** tab.

The steps to configure the connector are determined by the information you added to the connector template.

5 Under **Assertion Attribute Mappings**, map the SAML Assertion attributes to the appropriate attributes in your identity source.

6 Under **Trust policy for user identities in assertions**, configure the preferred action to take when the appliance receives an assertion from the identity provider:

- **Allow access for unknown users:** Creates unique user objects in an unmanaged internal identity store for the identity provider.

  For more information, see "Requirements for Using SAML2 In Identity Sources" on page 140.

- **Allow access for known users:** Matches user objects in managed identity sources.

For more information about unknown and known users, see "Understanding SAML2 In Identity Sources" on page 139.

7 Expand the **Federation Instructions**, then copy and paste the instructions into a text file to use when you configure the originating identity provider.

NOTE: You must use a text editor that does not introduce hard returns or additional white space. For example, use Notepad instead of Wordpad.

---

8 Click **OK** to save the configuration.

9 On the Admin page, click **Apply** to commit the changes to the appliance.

10 Wait until the configuration changes have been applied on each node of the CloudAccess cluster.

11 The appliance now acts as a SAML 2.0 service provider for the specified identity provider. The appliance SAML 2.0 metadata should now include the `SPSSODescriptor` section. Use this information to configure the identity provider for SAML 2.0 Inbound federation with the appliance.

12 Log in to the originating identity provider as the account administrator, then configure the SAML 2.0 Inbound federation for CloudAccess in the provider's administration console.

To complete the setup, use the information from the **Federation Instructions** in Step 7 and the `SPSSODescriptor` from Step 11.

NOTE: When you copy the appliance's signing certificate, ensure that you include all leading and trailing hyphens in the certificate's Begin and End tags.

---

13 (Conditional) If you enabled access for unknown users, you must configure entitlements for the users that will be added to the SAML2 In internal data store. In the CloudAccess administration console, click **Policy** on the toolbar, then perform policy mapping to specify entitlements for the SAML2 In users to the appropriate applications.

For more information, see "Mapping Authorizations" in the *NetIQ® CloudAccess and MobileAccess Installation and Configuration Guide*.

14 The appliance login page provides a link to the login page of the SAML 2.0 identity provider, located to the left of the user name and password login options. The SAML 2.0 users log in through the identity provider to gain access to the appliance landing page.

Basic SSO Custom Connectors

To import and configure a custom connector for Basic SSO:

1 Log in as an administrator to the CloudAccess administration console:

`https://appliance_dns_name/appliance/index.html`

2 Import the custom connector for Basic SSO to the **Applications** palette.

   2a Copy the custom connector ZIP file to the computer where you administer CloudAccess.

   2b Click the **Tools** icon on the console toolbar, then click **Import Connector Template**.

   2c Browse to and select the ZIP file for the connector template you want to import, then click **Import**.

   The imported connector appears in the **Applications** palette.

3 Drag the new custom connector from the **Applications** palette to the **Applications** panel.

   The Configuration window opens automatically for the initial configuration. To view or reconfigure the settings later, click the connector icon, then click **Configure**.

4 On the Configuration page, you can modify the display name, set custom settings as required, and view information about the connector's destination website.

5 Click the **Appmarks** tab, then review the default settings for the appmark.

   Public access is enabled automatically. If you disable public access, the appmark does not appear on the landing page until you map authorizations to set entitlements for user roles (groups).

6 Click **OK** to save the configuration.

7 On the Admin page, click **Apply** to commit the changes to the appliance.

8 Wait until the configuration changes have been applied on each node of the CloudAccess cluster.

9 (Conditional) If Public access is disabled, perform policy mapping to specify entitlements for identity source roles (groups).

   For more information, see "Mapping Authorizations" in the *NetIQ® CloudAccess and MobileAccess Installation and Configuration Guide*.

10 After you complete the configuration, users can log in through CloudAccess to access the destination website. The CloudAccess login page URL is:

   `https://appliance_dns_name`

7.14 Downloading and Importing Single Sign-On Connectors

NetIQ provides additional connectors for federated single sign-on. The connectors are available for download from NetIQ Downloads. You must import the connector into CloudAccess in order to use it.

To download and import a connector template to CloudAccess:

1 Download the template file for the desired connector:

   1a Launch a web browser on the computer that you use to manage CloudAccess.

   1b Go to NetIQ Downloads (https://dl.netiq.com/).

   1c Search for the desired connector, such as the connector for WebEx.

   1d Click the connector's download link and follow the instructions to download the file.

   1e When you are prompted, log in with your NetIQ customer account credentials, then continue with the download.

   1f Extract the connector's ZIP file and release notes to a local directory.

2. Log in as an administrator to the CloudAccess administration console:

   `https://appliance_dns_name/appliance/index.html`

3. Click the **Tools** icon on the toolbar, then click **Import connector template**.
4. Click **Browse,** then browse to and select the ZIP file that you extracted from the connector's download file.
5. Click **Import**.

   The **Applications** palette displays the connector that you imported.

6. Proceed to the configuration instructions for the desired connector.

After you import the connector, you must configure the connector settings in CloudAccess. For more information, see the requirements and configuration information for the desired connector.

8 Mapping Authorizations

Most companies define their business policies through authorization assignments. Examples of authorizations are groups, roles, and profiles. These authorizations are different depending on each SaaS application. For more information, see Section 8.1, “Supported Roles and Authorizations,” on page 151.

Authorizations give users access to resources. CloudAccess provides a simple solution that allows you to map your identity source roles (groups) to the SaaS application authorizations and approve or deny access to those authorizations.

Authorization categories are available for the connector types that provision users (Office 365, Google Apps, and Salesforce). If you use connector types that provide only authentication and they require mapped authorizations for entitlements instead of Public access, their authorizations are available in the Other Applications category. By default, Public access is disabled for all connectors, except for the connectors for Basic SSO.

The Policy Mapping page maps the authorizations from the SaaS applications to the roles (groups) in the identity sources and allows you to select whether the authorization requires an approval. If approval is required, the Approval page allows you to accept or deny the authorization request.

8.1 Supported Roles and Authorizations

Each identity source can contain different roles that appear on the Policy Mapping page.

- **Active Directory:** groups, local groups, and global groups
- **eDirectory:** group
- **Other Identity Sources:** roles for JDBC and SSUS identity sources

Each application contains different authorizations that appear on the Policy Mapping page.

- **Google Apps:** user and groups
- **Office 365:** account, groups, and license
- **Salesforce:** groups, roles, and profiles (account types)
- **Other Applications:** appmarks for Bookmarks, Simple Proxy, or other applications that require mapped authorizations for entitlements instead of Public access.

Provisioning applications support mapped authorizations only for users in Active Directory, eDirectory, and JDBC identity sources. For more information, see “Requirements for Provisioning” in the *NetIQ® CloudAccess Connectors Guide*.

8.2 Prerequisites

Verify that you meet the following prerequisites before mapping SaaS application authorizations to the identity source groups:

❒ Configure the appropriate connectors for your environment. For more information, see Chapter 7, “Configuring Connectors,” on page 77.

- ❒ Ensure that roles (groups) in the identity source exist.
- ❒ Populate the required attributes on the users in the identity source. For more information, see Section 4.1, “Verifying the Identity Source User Attributes,” on page 41.

8.3 Loading Authorizations

In order to map an authorization, you must load the authorization into the Policy Mapping page.

To verify that applications are available for mapping authorizations:

1. Verify that you have configured the SaaS application connectors that provision users.

   For more information, see Chapter 7, “Configuring Connectors,” on page 77.
2. Log in to the Admin page using the application administrator credentials you specified when you created the SaaS application connector.
3. Click **Policy** to open the Policy Mapping page.
4. In the right pane, click the down arrow next to the connector, then select your SaaS application connector, or select **Other Applications** and select the application.

   If the Policy Mapping page does not display the SaaS application connector, you did not configure the connector properly. For example, if the **Public** policy is enabled for a connector, the application does not appear in the list. For more information, see Chapter 7, “Configuring Connectors,” on page 77.

Successfully completing these steps populates the Policy Mapping page with the SaaS application’s authorizations.

8.4 Reloading Authorizations

When you perform a switch master with the cluster nodes, or if authorizations change in the SaaS applications, or you add new roles in the identity sources, you must reload the authorizations on the Policy Mapping page.

To reload authorizations:

1. To reload roles (groups) from the identity sources, click the **Reload table** icon at the end of the Identity Source table.
2. To reload authorizations from the SaaS applications, click the **Reload table** icon at the end of the Authorizations table.

8.5 Mapping Authorizations

After the authorizations load, map the SaaS application authorizations to the identity source roles (groups).

To map authorizations:

1. Log in with an appliance administrator account to the Admin page at `https://`*`appliance_dns_name`*`/appliance/index.html`.
2. Click **Policy** at the top of the page.

3 In the right pane of the Policy Mapping page, click the down arrow, then select the desired SaaS connector, or select **Other Applications** and select the application.

4 In the **Role Name** column on the left, select the role (group) from the identity source you want to map to an authorization from the selected SaaS connector.

5 In the right pane, drag and drop the desired authorization from the SaaS connector to the left mapping pane.

   or

   In the left pane, drag and drop the desired group from the identity source to the right mapping pane.

6 (Optional) Click the **Approvals** icon to specify that an approval is required to grant access.

   NetIQ recommends a maximum of 2,000 simultaneous approvals. For more information about approvals, see Section 8.8, "Approving Requests," on page 154.

7 Click **OK** to map the SaaS authorization to the identity source group.

The mapping grants access for users who are members of the identity source roles to the SaaS application authorization. When you add new users to the role (group) that is mapped to a SaaS account authorization, and the request is then approved (if approval is required), the users will see the associated appmark on the landing page or the MobileAccess application page. If **Prompt Before Provisioning** is not enabled, the accounts are provisioned automatically. If **Prompt Before Provisioning** is enabled (available for Salesforce and Google Apps only) users are prompted to create a new SaaS account or to claim an existing account the first time they click or tap the appmark. For information, see "How CloudAccess Provisions User Accounts" in the *NetIQ® CloudAccess Connectors Guide*.

8.6 Understanding Google Apps Mappings

Mapped placement of newly provisioned users to a sub-organization overrides the default placement in the top-level organization. On the Policy page, you can map the User Account authorization and a User Placement authorization value to the same identity source group. Once users are added to the appropriate identity source group, which triggers user account provisioning to Google Apps, new users are placed in the organization that you mapped to the identity source group in Policy Mapping.

IMPORTANT: Pay careful attention when mapping User Placement authorization values to identity source groups on the Policy page to ensure that users are placed in the intended Google Apps organizations. Google Apps allows each user to be placed in only one organization at a time. If you grant a User Placement authorization to a user, and then grant another User Placement authorization to the same user, the first value is overwritten when the user is moved in the Google Apps organizational unit structure.

In addition, if you revoke a User Placement authorization for a user, even if that user has multiple User Placement authorizations, that user is moved to the default organization specified in the Google Apps connector configuration. (Because Google Apps allows a user to be placed into only one organization at a time, when a User Placement authorization value is overwritten, there is only one value that is removed, which moves the user back to the default placement.) If you want to move that user from the default location back into a new Google Apps sub-organization, you must add the user back to the appropriate identity source group and perform policy mappings again.

If you revoke a User Account authorization for a user after the user has been provisioned into Google Apps and one or more User Placement authorizations have been granted to the user, that user is placed in suspended mode in Google Apps. No placement activity takes place as long as the user account is suspended; the user simply remains in the Google Apps organization that they were in

before the suspension. When you re-grant the User Account authorization, the account is moved to the “active user” state. If you performed a User Placement authorization change while the user account was suspended, the user is moved to the appropriate organization once the account is reactivated.

8.7 A Mapping Example

Use the following example of authorization mapping to understand how mapping works.

1. In Active Directory:
   - **1a** Create a group named GoogleAppsUsers.
   - **1b** Add users to the GoogleAppsUsers group.
2. In Google Apps for Business, create a Google Apps Account group.
3. In CloudAccess, configure the connector for Google Apps.
4. On the Policy Mapping page:
   - **4a** Load the authorizations for the connector for Google Apps.
   - **4b** In the **Role Name** column, select the connector for Active Directory.
   - **4c** In the right pane, select the connector for Google Apps.
   - **4d** Drag and drop the Google Apps Account into the left pane, over the GoogleAppsUsers group.
   - **4e** Click **OK**.

     The appliance automatically provisions all users in the GoogleAppsUsers group to the Google Apps Account group. Note that if the **Prompt Before Provisioning** option is enabled in the connector for Google Apps configuration, users are prompted the first time they log in to either create an account or specify an existing account.
5. In Active Directory, create a new user, then add the user to the GoogleAppsUsers group.

   The user is automatically added to the Google Apps Account group and has access to Google Apps for Business.

8.8 Approving Requests

CloudAccess provides the ability to approve or deny requests to the SaaS applications. During the configuration of the connector, you specified an application owner. The application owner approves or denies requests for access to the SaaS applications. The application owner knows who should have access to the SaaS applications, whereas the appliance administrator might not have this knowledge.

The **Approval** icon appears in the administration console only if you have mapped roles and selected the option to require approval for the account. When there are accounts waiting for approval, CloudAccess adds the **Approval** icon.

By default, CloudAccess automatically provisions users according to mapped authorizations. To enable approvals so that automatic provisioning does not occur, click the **i** (Configure Authorizations Policies) icon when you map the roles (groups) from the identity source to the SaaS applications authorizations on the Policy Mapping page. Now an application owner must grant approval before provisioning can occur.

To grant approval:

1. Log in to the Admin page at `https://appliance_dns_name/appliance/index.html` as the application owner.
2. Click the **Approvals** tab.
3. Select the desired approval request.
4. Click **Approve** or **Deny**.

NOTE: Users who have been deleted from the identity source may still appear on the Approval page. If you know that certain users have been deleted, you can simply deny approval for those users. However, approving requests for users who have been deleted does *not* result in account provisioning for those users in the SaaS applications.

NetIQ recommends a maximum of 2,000 simultaneous approvals.

9 Reporting

CloudAccess provides reports of users' activity through the appliance. You can run, download, and save various reports on the **Reports** tab in the administration console. CloudAccess also provides the option to use Google Analytics as an external dashboard, or to forward events to Sentinel Log Manager.

9.1 Using Google Analytics as an External Dashboard

CloudAccess enables administrators to use Google Analytics as an external dashboard to monitor and analyze CloudAccess usage. Once you have completed the free Google Analytics registration process for the CloudAccess appliance, data is available for analysis within a few hours. You can also do your own data mining with the API that Google provides. For more information, see the Google Analytics website (http://www.google.com/analytics/?gclid=CJCt792Y07kCFUlp7AodDBwALA).

To set up Google Analytics for CloudAccess:

1. (Conditional) If you do not already have a Google account, set one up on the Google web site.
2. Sign in to your Google account and select the option to register for Google Analytics.
3. Select the option to monitor a website and provide the base URL for the CloudAccess appliance. Google Analytics tracks both user and admin logins. For example, `https://appliance_dns_name`.
4. Specify an account name. This account name is only for managing Google Analytics and does not affect anything in CloudAccess. You can share this account name as needed.
5. Log in to the CloudAccess administration console.
6. On the Admin page, drag the Google Analytics icon from the **Tools** palette to the **Tools** panel.
7. Enter the Tracking ID (not the tracking code) that Google provided during the registration process and click **OK**.
8. Click **Apply** and wait for the appliance to update.

NOTE: If you have any issues with configuring the Google Analytics tool in the administration console, such as the tool being invisible on the Tools palette, verify that you do not have any ad blockers running in your browser that might be interfering with administration tasks. You should be able to disable any ad blockers on the web page itself.

9.2 Integrating with Sentinel Log Manager

The CloudAccess appliance can forward events to Sentinel Log Manager 1.2.x if you want more detailed reports.

To integrate the appliance with Sentinel Log Manager:

1. Configure Sentinel Link in Sentinel Log Manager.

   For more information, see Sentinel Link Overview Guide (http://www.novell.com/documentation/sentinel70/sentinel_link_overview/data/bookinfo.html).

2. Open TCP port 1290 on the Sentinel Log Manager server.
   - **2a** To change the port, use `ssh` and log in to the Sentinel Log Manager server as `root`.
   - **2b** At the command prompt, enter `yast firewall`.
   - **2c** Select **Advanced** > **Allowed Services**, then manually add port 1290 to the list of TCP ports.
3. On the Admin page in CloudAccess, drag and drop the Sentinel icon from the **Tools** palette to the **Tools** panel.
4. Click the Sentinel icon, then click **Configure**.
5. Specify the IP address and port of the Sentinel Link server, then click **OK** and **Apply** to save the changes.

The CloudAccess appliance appears as another event source in Sentinel Log Manager.

10 Configuring the End User Experience

CloudAccess allows you to configure the end user's email client or mobile devices to use the single sign-on authentication to access the SaaS applications. This increases the security of your company's information stored in the SaaS applications because users authenticate with their corporate credentials, but these credentials are never stored in the SaaS applications.

Configure each user's email client or mobile device to point to the CloudAccess appliance. The appliance acts as a proxy, so when users access the SaaS applications, the appliance automatically logs users in to the SaaS application.

CloudAccess also allows you to customize the login, logout, and landing pages so they display your company's branding instead of the default NetIQ branding.

10.1 Configuring Email Clients

You can configure any supported email client to point to CloudAccess. The email clients allow you to receive email from multiple sources in one location. For a list of supported clients, see "Email Clients" on page 16.

NOTE: The following procedure lists typical ports for email clients, but ports may vary depending on your environment.

To configure your email client to use a CloudAccess email account:

1. Access your email client.
2. Create a new email account using the following information to configure CloudAccess as your email source:

   Incoming email server (IMAP/POP): Specify the IP address or hostname of your appliance.

   Incoming email server username: Specify your identity source enterprise logon name for the account name.

   Incoming email server password: Specify your identity source password.

   If your password changes, you must change the password in the email account.

   Outgoing email server (SMTP): Specify the IP address or hostname of your appliance.

   SSL: You must select **SSL** for IMAP (port 993), POP (port 995), and SMTP (port 25).

   The SMTP server requires authentication.

   For more information, see the appropriate documentation for the email client you are using.

10.2 Configuring End User Browsers for Kerberos Authentication

If you are using Windows Integrated Authentication for Kerberos authentication to CloudAccess, each end user browser must be configured to use Kerberos authentication.

To configure Kerberos authentication for web browsers:

1. Add the user computers to the Active Directory domain.

   For instructions, see your Active Directory documentation.
2. Log in to the Active Directory domain, rather than the computer.
3. (Conditional) If you are using Internet Explorer, configure the browser to trust the appliance:
   - **3a** Click **Tools** > **Internet Options** > **Security** > **Local intranet** > **Sites** > **Advanced**.
   - **3b** In the **Add this website to the zone** field, enter the Base URL for the appliance, then click **Add**.

     In the configuration example, this URL is `serv1.cloudaccess.com`.
   - **3c** Click **Close**, then click **OK**.
   - **3d** Click **Tools** > **Internet Options** > **Advanced**.
   - **3e** Verify in the Security section that **Enable Integrated Windows Authentication** is selected, then click **OK**.
   - **3f** Restart the browser.
4. (Conditional) If you are using Firefox, configure the browser to trust the appliance:
   - **4a** In the URL field, specify `about:config`.
   - **4b** In the **Filter** field, specify **network.n**.
   - **4c** Double-click `network.negotiate-auth.trusted-uris`.

     This preference lists the sites that are permitted to engage in SPNEGO Authentication with the browser. Specify a comma-delimited list of trusted domains or URLs.

     For this example configuration, add `serv1.cloudaccess.com` to the list.
   - **4d** Click **OK**, then restart your browser.

10.3 Customizing Branding on User-Facing Pages

CloudAccess allows you to customize user-facing pages, such as the login page, so users see your company branding instead of the default NetIQ branding. After you have customized those pages, you can modify them as needed to meet new company requirements. Customizing the user pages does not affect any pages in the administration console itself.

If you implement custom branding in your CloudAccess environment and then re-run the initialization process to modify the DNS server or make other changes to an existing cluster, branding is reset to the default settings. Before you re-run the initialization process on an existing cluster, ensure that you back up your customized branding files so that you can reuse them.

IMPORTANT: Performing advanced branding customization requires advanced JavaServer Pages (JSP) knowledge. Before you make any changes, ensure that you have a good snapshot of your appliance that you can revert to if necessary. If you upload a bad branding file and are unable to log in to the administration console, you can re-run the appliance initialization to restore the default login pages. For more information, see Section 2.8, “Initializing the Appliance,” on page 27.

To customize branding for users:

1. (Conditional) If you plan to perform extensive rebranding, take a snapshot of the appliance.
2. Log in with an appliance administrator account to the administration console at `https://`*`appliance_dns_name`*`/appliance/index.html`.

3 On the toolbar, click the Tools icon, then click **End user branding**.

4 (Conditional) If you want to customize the user login page, complete the following steps:

   4a Click **Basic Customization**.

   4b Change the title and background colors by specifying HTML color codes.

   4c Change the default image by either not showing the image, or uploading a new image.

   The image size on the user landing and logout pages is always scaled to 60 x 60. On the login page, if the specified image is between a minimum of 60 x 60 and a maximum of 300 x 300, CloudAccess displays the image “as is.” If the image is outside this range, CloudAccess scales it to the nearest minimum or maximum value.

   4d Click **OK** to save the changes and then click **Apply**.

5 (Conditional) If you want to perform more extensive rebranding, complete the following steps:

   5a Click **Advanced Customization**.

   5b Click **Download default end user login code**.

   5c Save the file to your local computer.

   5d Save a backup copy of the file.

   5e Unzip the downloaded file and locate the `.jsp` files in the `osp\jsp` subdirectory.

   5f Modify the desired JSP pages. The default text for the login page is located in the `osp\resources\oidp_custom_resources_en_US.properties` file.

   For more information about the JSP files, see Appendix A, “Performing Advanced Branding,” on page 181.

   5g Zip up the files again, but include only the `images` and `jsp` directories.

   5h Log in to the CloudAccess console again.

   5i On the toolbar, click the Tools icon, then click **End user branding**.

   5j (Conditional) If you are customizing pages for the first time, click **Browse**, then browse to and select the modified file.

   5k (Conditional) If you are updating previously customized pages, delete the name of the existing file. Click **Browse**, then browse to and select the `.zip` file that contains the newly modified `.jsp` files.

   5l Wait until the file name changes to a hexadecimal value, then click **OK**.

   5m Click **Apply**.

   The pages now display the branding you customized in the `.jsp` files.

11 Maintenance Tasks

CloudAccess allows you to change various appliance configuration settings as needed. For example, moving your appliance from a staging configuration to a production environment requires changes to the networking components.

11.1 Changing the Cluster Password

You can change the administrator password for the cluster as needed. The administrator password is the same for all nodes in the cluster.

To change the cluster password:

1. On the Admin page, click the Cluster icon at the bottom of the page, then click **Change cluster password**.
2. Type your old password, then type your new password twice and click **OK**.

11.2 Configuring Session Timeouts

The admin session timeout is set to 10 minutes and is not configurable. The user session timeout is also set to 10 minutes by default, but it is configurable.

To change the user session timeout:

1. On the Admin page, click the Cluster icon at the bottom of the page, then click **Configure**.
2. Adjust the setting in the **User session timeout** field as needed, then click **OK**.

11.3 Changing the IP Address

You can change whether a node uses a DHCP IP address or a static IP address on the Admin page.

To change the IP address:

1. Click the node icon, then click **Configure**.
2. Select whether the appliance uses a DHCP IP address or a static IP address.

   If you select to use a static IP address, you can change the required values for the subnet mask, default gateway, and the DNS server.
3. Click **OK** to save the changes.
4. Click **Apply** to apply the changes to the appliance.

11.4 Changing the Public DNS Name or NTP Server Settings, or Uploading New Certificates

The appliance contains self-generated certificates. You can upload custom certificates through this interface. You can also change the public DNS name or NTP server if necessary.

1. Log in with an appliance administrator account to the Admin page at `https://appliance_dns_name/appliance/index.html`.
2. Click the cluster icon under **Appliances**, then click **Configure**.
3. Change the key pairs, NTP server, or public DNS name, then click **OK**.
4. Click **Apply** to apply the changes to the appliance.

Expired key pair certificates prohibit changes from being made to this page and make the key pair field red. If the key pair expires, you must re-initialize the appliance before you can upload a new certificate.

11.5 Updating the Appliance

CloudAccess provides an update channel for keeping your appliances current with the latest security fixes, bug fixes, and feature updates. Updates work only if you have registered each node in the cluster. For more information, see Section 3.3, “Registering the Appliance,” on page 30.

When an update is available for one or more nodes in the cluster, the CloudAccess Admin page displays a flag icon in the upper right corner of the window. You can also configure the appliance to send an email notification when an update is available. When you click the flag icon, you can see the version of the pending update, instructions on how to apply the update, and the Release Notes associated with the update.

The flag icon for the update channel appears only if you are logged in to the Admin page with an administrator account. Other consoles do not display the flag icon.

CloudAccess automatically checks the NCC channel for updates once daily at 11:23:23 p.m. and downloads any available update. You can also manually check for updates any time by clicking **Tools > Check for updates** on the Admin page. You can download and install an update as soon as the flag appears on the Admin page, or you can wait for CloudAccess to download the update that night, to minimize network impact due to possible size of an update.

WARNING: If you download and update in the same step and the download is interrupted or incomplete, the update fails. The appliance might become unresponsive or seem to be in a restart loop. If this occurs, download the update, then go back to the snapshot and try again to apply the update.

NetIQ recommends always keeping your appliance up to date. However, updates are cumulative, so if you miss an update you can just install the next one when it is available.

IMPORTANT: If you apply an update to one node, you must apply the update to all the other nodes in the cluster. Update one node at a time. Ensure that the update was successful and the node is still working properly before you begin updating the next node. Do not perform any other administrative tasks requiring an **Apply** command, and do not switch the master node, until all nodes have been successfully updated to the same version of CloudAccess.

This process allows you to run in a mixed environment while updating each node. Once you have applied all available channel updates, the flag icon goes away.

To apply an update:

1. Take a snapshot of each node in the cluster to create a backup.
2. Click the desired node, then click **Apply update**.

   CloudAccess displays status messages during the installation of the update and the rebooting of the node.
3. After the update completes and the node restarts, click **About** on the node to verify the updated version.
4. Verify the health of the updated node and all of the nodes in the cluster. Make sure all icons are green.

   For more information, see Section 12.2, “Displaying Health,” on page 167.
5. Repeat Step 2 through Step 4 for each node in the cluster.
6. When you are sure all of the nodes in the cluster are working as expected, delete the snapshot.

11.6 Shutting Down or Rebooting a Node

You can shut down or reboot a node in the cluster if necessary.

NOTE: If you shut down the node in a single node cluster, the administration console becomes inaccessible. You must then use vSphere to power on the node. Similarly, if you reboot the node in a single node cluster, the administration console is inaccessible until the reboot is complete.

To shut down or reboot a node:

1. Log in with an appliance administrator account to the Admin page at `https://`*`appliance_dns_name`*`/appliance/index.html`.
2. Click the node that you want to shut down or reboot, then click **Shutdown/Reboot** on the menu.
3. In the confirmation window, click **Shutdown** or **Reboot**.
4. (Conditional) Wait for the node to reboot, or use vSphere to power the node back on.

11.7 Recovering from a Disaster

Use snapshots of the nodes to recover from a disaster. It is important to take snapshots of each node in the cluster regularly so you do not lose information.

To recover from a disaster:

1. On a regular basis, take snapshots of the nodes in the cluster.

   1a Power off the working node, then take a snapshot. NetIQ recommends this method, but it requires that you shut down and restart the node in order to take the snapshot.

   or

Take a snapshot of the running node, ensuring that you include the virtual machine's memory. Including the memory in the snapshot requires more time and space to store the snapshot, but taking a snapshot of a running node without the memory can result in corruption.

1b Repeat Step 1a for each node in the cluster, within a short time.

2 When a failure happens, restore the master node snapshot first.

3 Restore the other nodes in the cluster.

Use these steps only for disaster recovery. Never restore one snapshot. CloudAccess contains a database that is time-sensitive. Restoring one node only and not the others causes corruption in the appliance.

12 Troubleshooting CloudAccess

The CloudAccess administration console displays health status information for the system, the nodes, and the cluster on the Admin page. This section describes the health status indicators, how to troubleshoot health issues, and how to work around known issues.

12.1 Troubleshooting the Appliance Initialization

If the appliance initialization fails, the user interface displays a link for log files. Click the link to download the log files that provide information about the failure.

12.2 Displaying Health

CloudAccess displays health status information for each node and for the cluster at the bottom of the Admin page. Hover the mouse over each node to display the health status of the node. If you want more details, click the node, then select **Show Health.** CloudAccess refreshes health status information every five minutes.

When you click **Show Health**, CloudAccess displays the status for each component of the appliance. If the status is anything other than green (healthy), use the troubleshooting tools to determine what is wrong.

12.3 Using Troubleshooting Tools

CloudAccess provides troubleshooting tools to help you resolve problems.

To access these tools:

1. Log in with an appliance administrator account to the Admin page at `https://`*`appliance_dns_name`*`/appliance/index.html`.
2. Under **Appliances**, click the node icon, then click **Enter troubleshooting mode**.
3. Click the node icon again, then click **Troubleshooting tools**.
4. Select one or more of the troubleshooting scenarios listed.
5. Duplicate the error or condition.
6. Click **Download CloudAccess Log Files** to download the logs.

After you obtain the logs, turn off troubleshooting mode by clicking the node icon again and then clicking **Exit troubleshooting mode**. Leaving the logs running affects the performance of your appliance.

All of the log files in Table 12-1 are included in the download, no matter what scenario you select. The scenario you select determines the amount of data displayed in the log files. Search the appropriate log file for errors while troubleshooting issues.

Table 12-1 Troubleshooting Log Files

Feature	Logs
Initialization or commands	`ConfigurationReplicator.log`
	`ConfigurationReplicator_RL.log`
	`messages`
	`boot*`
	`packageoperations.log`
	`dserv.log`
	`firewall`
Forward proxy	`access.log`
Admin.html UI	`adminui.log`
Registration	`register.log`
Updates	`zypper.log`
	`downloadUpdate.log`
	`afterUpdate.log`
	`beforeUpdate.log`
	`rpmsAfterUpdate.log`
	`rpmsBeforeUpdate.log`
	`rpmsUpdateDiff.log`
	`300_appliance_SnapshotUconPackages.sh.log`
Identity Source Provisioning	`bis_AD_<xxxxx>.log`
	`bis_AD_<xxxxx>_RL.log`
	`ConnectorLogs.txt`
	`bis_EDIR_h2q3p.log`
	`bis_EDIR_h2q3p_RL.log`
Provisioning to the SaaS Applications	`connectors_SFORCE_<xxxxx>_RL.log`
	`connectors_GOOGLEAPPS_<xxxxx>.log`
	`connectors_GOOGLEAPPS_<xxxxx>_RL.log`
	`connectors_O365_<xxxxx>.log`
	`connectors_O365_<xxxxx>_RL.log`
	`ConnectorLogs.txt`
Mapping	`RolesandResourceServiceDriver.log`
	`UserApplicationDriver.log`

Feature	Logs
Approvals	`jboss.log`
Reporting	`ManagedSystemGatewayDriver.log`
	`DataCollectionServiceDriver.log`
Mobile Devices	`mail`
	`mail.err`
	`mail.info`
Custom Connectors	`catalina.out`
End User Authentication	`catalina.out`

12.4 Troubleshooting Different States

CloudAccess displays indicators for the current state of the different appliance components. The display refreshes every five minutes. CloudAccess might not immediately display the change.

The following sections list the different components, the possible states, and troubleshooting steps you can take when the state changes.

- Section 12.4.1, “Master Node Health,” on page 169
- Section 12.4.2, “Front Panel of the Node,” on page 169
- Section 12.4.3, “Top of the Node,” on page 170
- Section 12.4.4, “Identity Source,” on page 170
- Section 12.4.5, “Applications,” on page 171
- Section 12.4.6, “Tools,” on page 172

12.4.1 Master Node Health

The master node is responsible for all administration functions in CloudAccess. If the master node is not running, the following functions do not work: provisioning or deleting user accounts, mapping authorizations, system roles, approvals, and reporting. Other nodes in the cluster continue to capture and cache events, but do not send those events to the master node until it is running again. Similarly, event forwarding to Sentinel does not work as long as the master node is down.

12.4.2 Front Panel of the Node

The indicator on the front panel of the node displays the health state of the node.

Figure 12-1 *Front Panel*

The states are:

Green: The node is healthy.

Yellow: The node cannot communicate with the other nodes within the five minute refresh.

Red: The node cannot communicate with the other nodes within two of the five minute refresh cycles.

Clear: The node is initializing or the state of the node is unknown.

Perform the following troubleshooting steps in the order listed if the state is anything but green:

1. Wait at least five minutes for the display to refresh and display the current state.
2. Click the node, then select **Show health**.

   Show Health displays which part of the appliance is having issues.
3. If Show Health displays a problem, use the troubleshooting tools to gather logs.

   For more information, see Section 12.3, “Using Troubleshooting Tools,” on page 167.
4. Restart the appliance, then wait at least another five minute cycle for all nodes to display the current state.

12.4.3 Top of the Node

The indicator on the top of the node shows whether the **Apply** commands completed successfully.

Figure 12-2 *Top of the Node*

The states are:

Green: All **Apply** commands completed successfully.

Red: The **Apply** commands did not complete successfully.

Perform the following troubleshooting steps in the order listed if the state is red:

1. Mouse over the top of the node to see the status of the last **Apply** command made on the node.
2. If there is not enough information in the summary, click **Enter troubleshooting mode** on the node, then mouse over the node again.

   The troubleshooting mode displays a details summary of the last **Apply** command made on the node.
3. Restart the appliance, then wait at least another five minute cycle for all nodes to display the current state.

12.4.4 Identity Source

The health indicator for the identity source is the small icon in the lower left corner.

Figure 12-3 Identity Source Indicator

The states are:

Green: The connector to the identity source is healthy.

Yellow: The connector has communication problems with the identity source.

Red: The connector to the identity source is unhealthy or contains errors.

Question mark: The state of the connector to the identity source is unknown.

Perform the following troubleshooting steps in the order listed:

1. If the connector is green, but the CloudAccess interface is not displaying users, verify that the identity source servers are running and communicating properly.
2. Use the troubleshooting tools to gather logs, then look at the identity source provisioning logs listed in Table 12-1 on page 168 for errors. The `ConnectorLogs.txt` file maps the display name of the connector with the log name of the connector, if there is more than one identity source connector.
3. Click **Show health** on the master node, then expand **Operational**.

   If these items are yellow or red, the interface displays helpful information to help troubleshoot the issue.
4. If you are using LDAPS to communicate with the identity source, verify the LDAP certificates are not expired. You refresh the certificates as follows:
   a. Log in to the Admin page, then click **Configure** on the identity source.
   b. Click the **Refresh** icon next to the identity source server.

12.4.5 Applications

The health indicator for an application connector is the small icon in the lower left corner.

Figure 12-4 Application Indicator

The states are as follows:

Green: The connector to the application is healthy.

Yellow: The connector to the application contains warnings.

Red: The connector to the application contains errors or cannot communicate with the application.

Question mark: The connector to the application is in an unknown state.

Perform the following troubleshooting steps in the order listed:

1. Click **Show health** on the master node, then expand **Operational**, and check the status of **Provisioning**.

   If **Provisioning** is yellow or red, CloudAccess displays helpful information to help troubleshoot the issue.
2. Use the troubleshooting tools to gather logs, then look at the provisioning logs listed in Table 12-1 on page 168 for errors.
3. Make a cosmetic change to the application connector configuration, then click **Apply**.

   By forcing an **Apply**, the appliance refreshes the application connector state and this can resolve the issue.

12.4.6 Tools

The health indicator for a tool is the small icon in the lower left corner. Only tools that report health have an indicator. The following tools do not have a health indicator: Google Analytics, Mobile, and Time-Based One-Time Password (TOTP).

Figure 12-5 *Tool Indicator*

For all tools, the **Question Mark** icon indicates that the tool is in an unconfigured state.

Advanced Authentication: The states for the Advanced Authentication tool are as follows:

- **Green circle:** The connection to the NetIQ Advanced Authentication Framework is healthy, and only Active Directory identity sources exist in the configuration.
- **Yellow triangle:** The connection to the NetIQ Advanced Authentication Framework is healthy. The triangle indicator serves as a warning that identity source types other than Active Directory exist in the configuration and are not supported with the Advanced Authentication Framework authentication providers.
- **Red circle:** The connection to the NetIQ Advanced Authentication Framework is not working. The Advanced Authentication Framework server is unreachable.

Authentication Filter: The states for the Authentication Filter tool are as follows:

- **Green circle:** The connection to the destination ExtAPI script uses HTTPS. The traffic is secure.
- **Yellow triangle:** The connection to the destination ExtAPI script uses HTTP. The traffic is not secure.
- **Red circle:** The connection to the destination ExtAPI script is not working. The ExtAPI script is unreachable.

Forward Proxy: The states for the Forward Proxy tool are as follows:

- **Yellow triangle:** The connection to or through the proxy is healthy. The triangle indicator serves as a warning that use of Forward Proxy is intended for test environments only.
- **Red circle:** The connection to or through the proxy is not working. The proxy device is unreachable.

Google reCAPTCHA: The states for the Google reCAPTCHA tool are as follows:

- **Green circle:** All of the configured identity sources are valid for use with reCAPTCHA.
- **Yellow triangle:** One or more of the configured identity sources are not valid for use with reCAPTCHA. For more information, see Section 5.2.1, “Requirements for reCAPTCHA,” on page 51.
- **Red circle:** None of the configured identity sources are valid for use with reCAPTCHA.

Sentinel and Syslog: The states for the Sentinel and Syslog tools are as follows:

- **Green circle:** The connection to the specified address:port is healthy.
- **Red circle:** The connection to the specified address:port is not working.

12.5 Troubleshooting Networking Issues

As an appliance administrator or network administrator, you may need to troubleshoot some basic networking issues before you can successfully initialize the CloudAccess appliance. For example, if your appliance boots onto the network and gets the wrong IP address or falls back to the default IP address, you can use a basic set of commands in the network troubleshooting console to help you resolve these issues.

The console is in a `chroot` jail environment that gives you temporary connectivity to the appliance. You can use the `cat` command to check files such as `/etc/resolv.conf` and `/etc/hosts`, which exist in memory in the `chroot` environment. When you run the initialization process, the real files are updated on the system. Similarly, actions such as updating the IP address or route are not persistent and are reset if you reboot the appliance. You must run through the initialization process to set them permanently.

NOTE: As long as you have not yet completed the initialization process on the appliance, you automatically have console access without login credentials. Logins to the troubleshooting console are no longer available once you have completed the initialization process. However, you can change network settings after this point using the Init screens.

To troubleshoot network issues:

1. Using the vSphere client console or a similar tool, access the troubleshooting console.
2. (Optional) Press the Tab key twice to see all available commands. Some commonly used and supported commands are listed below.
3. (Conditional) Use the following steps if you need to change appliance network settings:
   - **3a** Delete the default route: `route del default`
   - **3b** Create an alias with the IP address that you want to use: `ifconfig eth0:0` *`IP_address`* `netmask` *`subnet`*
   - **3c** Enter `ifconfig` again to verify that the address is now available.
   - **3d** Add a route: `route add default gw` *`gateway_IP`* `eth0:0`

3e Check connectivity to various resources: for example, `ping gateway_IP` or `ping www.google.com`

3f (Conditional) If DNS name resolution is not functioning correctly, add an entry to the `/etc/resolve.conf` file as follows: `echo "nameserver 0.0.0.0" > /etc/resolv.conf`

At this point, you should have connectivity to the appliance. You can run through the initialization process and configure the appropriate settings permanently.

3g In a browser, enter `https://IP_address/appliance/Init.html`, replacing *IP_address* with the IP address that you used for your alias.

3h At the certificate warning prompt, add an exception.

3i In Step 1 - Network of the initialization process, replace the default network values with your preferred IP address and other network settings and click **Next**.

3j After CloudAccess validates your entries, click **OK** to apply the new settings. CloudAccess applies the settings permanently and restarts services as needed.

If you return to the troubleshooting console, it now displays your preferred IP address. You can use the `ifconfig` command to verify that the new settings are working correctly.

3k Continue with the remaining initialization steps.

Supported commands include the following:

- `arp`
- `bash`
- `cat`
- `date`
- `echo`
- `ifconfig`
- `ip`
- `mkdir`
- `netcat`
- `nslookup`
- `ping`
- `pwd`
- `rm`
- `route`
- `sntp`
- `traceroute`

The following table provides examples of some common actions and commands.

Table 12-2 *Examples*

Action	Command
Set the IP address of the appliance.	`ifconfig eth0 static_IP netmask netmask up`
Delete the default route.	`route del default`
Set the default route (or gateway).	`route add default gw gw_IP eth0`

Action	Command
Update the time.	`sntp -P no -v -r pool.ntp.org`
Check networking.	`ping 8.8.8.8`
	`traceroute 8.8.8.8`
Verify that DNS is working.	`nslookup www.google.com`

12.6 Troubleshooting Provisioning Issues

Use the information in the following sections to troubleshoot provisioning issues.

12.6.1 Capturing Logs for Provisioning Issues

By default, when you provision users, the appliance does not log any information about the provisioning process. Enabling logging causes performance issues, so you should never leave the logging running all of the time.

However, if you have provisioned users and select accounts are not showing up in the SaaS applications, you will need to enable logging to capture the cause.

To capture logs of provisioning issues:

1. Determine which account (or accounts) were not provisioned.
2. Remove that account from the group that grants access to the SaaS application.
3. Turn on logging:
   - **3a** Access the Admin page of the appliance console.
   - **3b** Under **Appliances**, click the **Node** icon, then click **Enter troubleshooting mode**.
   - **3c** Click the **Node** icon again, then click **Troubleshooting tools**.
   - **3d** Select **Provisioning**, then click **Apply** and **OK**.
4. Add the user to the group or initiate the provisioning action.
5. Access the troubleshooting tools again.
6. Click **Download CloudAccess Log Files** to download the logs.
7. Deselect **Provisioning**, then turn off troubleshooting mode.
8. Review the logs to find the error when provisioning occurs.

Ensure that you have turned off troubleshooting mode, otherwise it will cause performance problems with the appliance.

12.6.2 Understanding the Result of Actions Performed on Users and Groups

Actions that are taken on users and groups in the identity source might not be reflected in the SaaS applications (Google Apps, Salesforce, and Office 365). The following table lists the actions in the identity sources and the corresponding actions in the SaaS applications.

Table 12-3 Provisioning Actions

Identity Sources	SaaS Applications
Delete a user. (Or disable a user account.)	Disables the SaaS account. **NOTE:** In the MobileAccess app an iOS device, the user continues to have access to the SaaS account until the in-progress user session times out.
Remove a user from the authorized group.	Disables the SaaS account.
Create a user.	◆ Creates an account for the user in the SaaS application, if the user is a member of a group with mapped SaaS authorizations. or ◆ Users are prompted to validate their information when they log in the first time.
Move a user from out of the search context into the search context.	Creates an account for the user in the SaaS application, if the user is a member of a group with mapped SaaS authorizations.
Move a user out of the search context.	Disables the SaaS account.

By default, CloudAccess establishes identity based on an internal unique ID in the identity source, not based on the user name, and does not support recreating users with the same name unless they also have the same internal unique ID. Once a user has been mapped and provisioned, if you delete the user from the identity source and then recreate that user with the same name, you will not be able to cache and activate the user in CloudAccess or provision the user to SaaS applications. When CloudAccess is unable to cache users properly, the Cached User Status Bar indicates this status with a lower number of active users than cached users.

IMPORTANT: CloudAccess does provide a **Relaxed user matching** option under **Advanced Options** on the configuration window for the identity source. If you select this option, CloudAccess matches users based on CN or sAMAccountName instead of the internal unique ID. This option enables you to recreate previously deleted users so CloudAccess can manage them again, but you must ensure that you do not create different users with the same CN or sAMAccountName as previously deleted users. Otherwise, those users will have access to the previously deleted users' cloud application data.

12.7 Troubleshooting Policy Mapping Issues

Use the information in the following sections to troubleshoot policy mapping issues.

12.7.1 No Connectors Appear on the Policy Mapping Page

If the Policy Mapping page does not display the connectors for the SaaS applications, there are two possible solutions:

- Verify that the connectors are configured properly and enabled. For more information, see the appropriate sections for configuring connectors in the *NetIQ CloudAccess Connectors Guide*.
- Click the **Refresh List** icon in the upper-right corner of the Policy Mapping page.

12.7.2 CloudAccess Does Not Reconcile Pending Approvals with Changes to Policy Mappings

Issue: CloudAccess does not reconcile pending approvals with changes to policy mappings. Users with pending approvals are granted the pending requests even if the mappings were removed after the requests were launched.

Workaround: If a policy mapping for a resource occurs by mistake, decline all the requests for that resource. If a policy mapping for a resource occurs correctly, but then the mapping is removed, simply decline all outstanding approval requests. You can often avoid this issue by ensuring that requests are approved or denied in a timely manner.

12.7.3 Using Multiple Browsers or Browser Windows Can Result in Duplicate Mappings

Issue: If you simultaneously use more than one browser or browser window to map authorizations, CloudAccess does not warn you if you inadvertently do the same mapping in two different browsers. Clicking **Refresh** displays two identical mappings on the Approvals page, but only one of them is a valid mapping. If you remove one of the mappings, CloudAccess might not actually deprovision the user until you remove the authorization that is mapped to the group.

Workaround: You can avoid this issue by using only one browser when you create policy mappings. To work around this issue, on the CloudAccess Policy page, manually remove all duplicate authorization mappings from the role, then map the desired authorizations back to the role.

12.8 Troubleshooting Mobile Device Issues

12.8.1 Safari on Mobile Devices Cannot Access the Login Page After You Enable the MobileAccess Connector

If users cannot access the CloudAccess login page from the Safari browser on a mobile device after you enable the MobileAccess connector in the administration console, verify that they have installed the MobileAccess app on their mobile devices. This is a requirement once you have enabled the MobileAccess connector in the administration console.

12.8.2 Device Registration Issues

The user might not be able to register a mobile device if the device is connected to a network that uses HTTP proxy. The user receives the following error message: `Unable to parse metadata.` To work around this issue, the user can register the device from a different network that does not use HTTP proxy.

12.9 Troubleshooting CloudAccess Login Failures

When a user is unable to log in to the CloudAccess appliance, the login page displays the message `Login failed, please try again`. Some causes of a login failure might not be obvious to either a user or a CloudAccess administrator, such as when multiple users have the same email address.

To help administrators determine the cause of the failure, the error message provides mouseover text containing an ID. You can download the `/var/log/tomcat7/catalina.out` log file and then search the log file for the ID provided in the error message. For more information about downloading log files, see Section 12.3, “Using Troubleshooting Tools,” on page 167.

12.10 Troubleshooting Authentications or Single Sign-On Issues

There can be multiple reasons why authentications to the SaaS applications (Google Apps, Salesforce, and Office 365) fail.

Time Synchronization: CloudAccess depends on timestamps to function correctly. Synchronize time between the VMware host, the appliance, and the workstations. Download the authentication or single sign-on logs. In the `catalina.out` file, search for the error `clock skew`.

SAML Authentications: Firefox contains a SAML debug add-on you can use to view the SAML authentication between CloudAccess and the SaaS applications. Download the add-on SAML tracer (https://addons.mozilla.org/en-US/firefox/addon/saml-tracer/) to view the SAML request.

Master Node Down: If the master node is not running, users who already have accounts can log in to SaaS applications, but CloudAccess cannot provision new users. So, if new users attempt to log in to SaaS applications and receive an error indicating they should contact their system administrator, verify that the master node is running.

12.11 Troubleshooting Connector Issues

For information about viewing the current health status of a connector, see Section 12.4.5, “Applications,” on page 171.

For information about accessing log files for connectors, see Table 12-1, “Troubleshooting Log Files,” on page 168.

For information about troubleshooting a specific connector, see the connector’s information in the *NetIQ® CloudAccess Connectors Guide*.

12.12 Troubleshooting JDBC Identity Source Issues

If you are having problems with your JDBC identity source, use the following information.

Question: I have copied all of my users into the indirect.usr table, the **user count** on my appliance Admin page has increased, but my users cannot log in. The **user count** never reconciles to the same number.

I also see one of the following messages in the `catalina.out` file: `mssqljdbc count: 20`, or `oraclejdbc count: 20`.

Answer: The connector for JDBC uses triggered publications, meaning the connector will only cache the users in the indirect.usr table if some event (insert, update, delete) has occurred on the indirect.usr table entry, so that the built-in triggers will add the rows to the indirect.indirect_process table for the connector to consume and process.

To trigger a synchronization of the existing users in the indirect.usr table, you must perform a trigger action. For example, run an SQL query that would touch the records and trigger an update.

```
UPDATE indirect.usr SET disabled = disabled WHERE idu IS NOT NULL
```

Question: I cannot seem to use a user from my real user database tables as an appliance administrator when I run the initialization process for the appliance. The administrator user name keeps coming back invalid and turning red, even though the user exists in my database login table.

Answer: The connector for JDBC only connects directly to the indirect tables. The issue is that the administrative user who you wish to use during the initialization process has not been copied from the real user database tables to the indirect.usr table. `Copy/select into` the desired administrative user to the indirect.usr table.

Question: I have changed the `indirec.proc_authuser` stored procedure to use my desired tables for authentication, but I am still getting a login failure.

Answer: There are two separate ways that the stored procedure verifies the user name and password parameters passed to the `proc_authuser` stored procedure.

- **Oracle:** The password parameter is hashed and then compared to the existing password with the default settings.
- **MSSQL:** The stored procedure uses the `PWDCOMPARE` built-in function with the default settings.

In some cases, when customizing the stored procedure, the password might not be encrypted. In cases like this, the `password=` clause in the stored procedure might also need to be altered.

A Performing Advanced Branding

CloudAccess allows you to customize user-facing pages, such as the login page, so users see your company branding instead of the default NetIQ branding.

IMPORTANT: Performing advanced branding customization requires advanced JavaServer Pages (JSP) knowledge. Before you make any changes, ensure that you have a good snapshot of your appliance that you can revert to if necessary. If you upload a bad branding file and are unable to log in to the administration console, you can re-run the appliance initialization to restore the default login pages. For more information, see Section 2.8, “Initializing the Appliance,” on page 27.

The brandable code is contained in a set of JSP files in the `/osp/jsp` directory.

To change the standard header or footer:

Edit the HTML formatting in one or both of the following files: `inc_common_body_top.jsp` or `inc_common_body_bottom.jsp`. The Java variables that the `inc_*.jsp` requires are listed at the top of the `inc_*.jsp` file. Much of this work can be done using the **Basic Customization** feature in the CloudAccess administration console. For more information, see Section 10.3, “Customizing Branding on User-Facing Pages,” on page 160.

To change the body (the space between the header and footer) of a page:

Each login, logout, and landing page contains the HTML formatting code that draws the “body” of its respective page. To customize this section of the page, you can edit the HTML formatting located between the include statements for the header and footer.

A.1 //Shared Include Files

inc_common_imports.jsp

Included in all non-shared JSPs in the Java imports area. Contains all shared import statements.

inc_common_java.jsp

Included in all non-shared JSPs in the Java code area. Contains the main Java code that processes request parameters and gathers data into standard Java variables that are used by the shared JSPs.

inc_common_head.jsp

Included in all non-shared JSPs in the JavaScript `<script>` area. Contains shared JavaScript functions that are used by the shared JSPs.

inc_common_body_top.jsp

Included in all non-shared JSPs at the top of the HTML `<body>` area. Contains the HTML formatting that creates the header for all non-shared pages. The header generally contains the product logo and name.

inc_common_body_bottom.jsp

Included in all non-shared JSPs at the bottom of the HTML `<body>` area. Contains the HTML formatting that creates the footer for all non-shared pages. The footer generally contains the NetIQ logo and a “demo version expired” warning, if applicable.

inc_common_locale.jsp

Included in the `loginselect.jsp` file in the Java code area. Contains Java code that sets the “temporary locale” form data items.

inc_common_usermessages.jsp

Included in all non-shared JSPs in the middle of the HTML `<body>` area. Contains the HTML formatting that creates the user error message section for all non-shared pages.

A.2 //Standard Login Page

loginselect.jsp

Contains the standard name-password form-based authentication page. All configurations of this page follow the standard header, body, and footer convention where the header and footer are defined in shared `inc_*.jsp` files. There are several optional objects on this page: radius login, recaptcha, and authentication contract selection.

A.3 //Second Factor Authentication Login Pages

logintotp.jsp

Contains the standard Timed One Time Password (TOTP) form-based authentication page.

fidoregister.jsp

Contains the standard Fast Identification Online (FIDO) form-based registration page.

fidologin.jsp

Contains the standard Fast Identification Online (FIDO) form-based authentication page.

A.4 //Landing Page (Home Page after Login)

landingpage_select.jsp

Contains the standard home page where users can select appmarks.

A.5 //Logout Page

landingpage_loggedout.jsp

Contains the “successful logout” page where the user is asked to close the browser to complete the logout.