



# Firewall Authentication in **FortiGate**

# After completing this document, you will be able to achieve these objectives about FortiGate Methods of Firewall Authentication:

- **What is the Firewall Authentication?**
- **FortiGate Methods of Firewall Authentication**
- **Local Password Authentication**
- **Server-Based Password Authentication**
  - Different Types of Server-Based Password Authentication
  - What is LDAP in detail?
  - LDAP Structure
  - Configuring an LDAP Server on FortiGate
  - What is RADIUS?
  - Configuring a RADIUS Server on FortiGate
  - Testing the LADP & RADIUS Query on the CLI
- **What is Two-Factor Authentication?**
  - Different Types of Two-Factor Authentication?
  - What is FortiToken?
  - FortiToken in Detail
  - Assigning a FortiToken to a User
  - FortiToken MFA process
  - What is the FortiToken Mobile?
  - Registering, Provisioning, and Activating FortiToken Mobile in detail
- **Authentication Methods**
  - What is an Active Authentication?
  - What is a Passive Authentication?
  - Applying Firewall Authentication to a Firewall Policy – Source
  - Protocols in Authentication when Applying to a Firewall Policy
  - Applying Firewall Authentication to a Firewall Policy – Source
  - Mixing Policies
  - Solutions to force FortiGate to send Login Prompt to the users
  - What is the Active Authentication Behavior?
  - What is the Captive Portal?
  - Configuring Captive Portal on a Network Interface
- **AuthenTication Timeout and Configuration**
- **Monitoring Authenticated Users**
- **LAB 1: Configuring an LDAP Server on FortiGate**
- **LAB 2: Configuring a RADIUS Server on FortiGate**

# Firewall Authentication

Traditional firewalling grants **network access** by verifying **the source IP address and device-type**. This is inadequate and can pose a security risk because the **firewall cannot determine who is using the device to which it is granting access**.

FortiGate includes authentication of **users** and **user groups**. As a result, you can follow individuals across multiple devices.

Where access is controlled by a user or user group, users must authenticate by entering valid credentials (such as username and password). **After FortiGate validates the user, FortiGate applies firewall policies and profiles to allow or deny access to specific network resources**.

## Firewall Authentication

- Includes the authentication of users and user groups
  - It is more reliable than just IP address and device-type authentication
  - Users must authenticate by entering valid credentials
- After FortiGate identifies the user or device, FortiGate applies firewall policies and profiles to allow or deny access to each specific network resource



**Authentication Required**

Please enter your username and password to continue.

Username

Password

**Continue**

# FortiGate Methods of Firewall Authentication

## FortiGate Methods of Firewall Authentication

- Local password authentication
  - Username and password stored on FortiGate
- Server-based password authentication (also called remote password authentication)
  - Password stored on a POP3, RADIUS, LDAP, or TACACS+ server
- Two-factor authentication
  - Enabled on top of an existing method
  - Requires something you know and something you have (token or certificate)

**FortiGate supports multiple methods of firewall authentication:**

- **Local password authentication**
- **Server-based password authentication (also called remote password authentication)**
- **Two-factor authentication** → This is a system of authentication that is enabled on top of an existing method—it cannot be enabled without first configuring one of the other methods. It requires something you know, such as a password, and something you have, such as a token or certificate.

**During this document, you will learn about each method of firewall authentication in detail.**

# Local Password Authentication

The simplest method of authentication is **local password authentication**. User account information (username and password) is **stored locally on the FortiGate device**. This method works well for a single FortiGate installation.

Local accounts are created on the **User Definition** page where a wizard takes you through the process. For local password authentication:

- Select **Local User** as the user type and
- Create a username and password

If desired, you can also add email and SMS information to the account, enable two-factor authentication, and add the user to a pre-configured user group.

After you create the user, you can add the user—or any preconfigured user group in which the user is a member—to a firewall policy, in order to authenticate.

You will learn about user groups and firewall policies in this lesson.

The screenshot illustrates the Local Password Authentication process on a FortiGate device. It consists of three main sections: a diagram, a top navigation bar, and a central configuration window.

**Diagram:** On the left, a user icon (1) is connected by a red arrow to a "FortiGate" device icon. A second red arrow (2) points from the FortiGate icon to a "User name and password" input field. This visualizes the flow from user input to the FortiGate system.

**User & Authentication > User Definition:** This section shows the "User & Authentication" path selected in the top navigation. Below it, the "User Definition" sub-path is highlighted. A progress bar indicates steps 1 (User Type) through 4 (Extra Info). Step 1 is marked with a green checkmark and "Local User" is selected. Steps 2, 3, and 4 are also marked with green checkmarks.

**Central Configuration Window:** This window contains four tabs corresponding to the wizard steps:

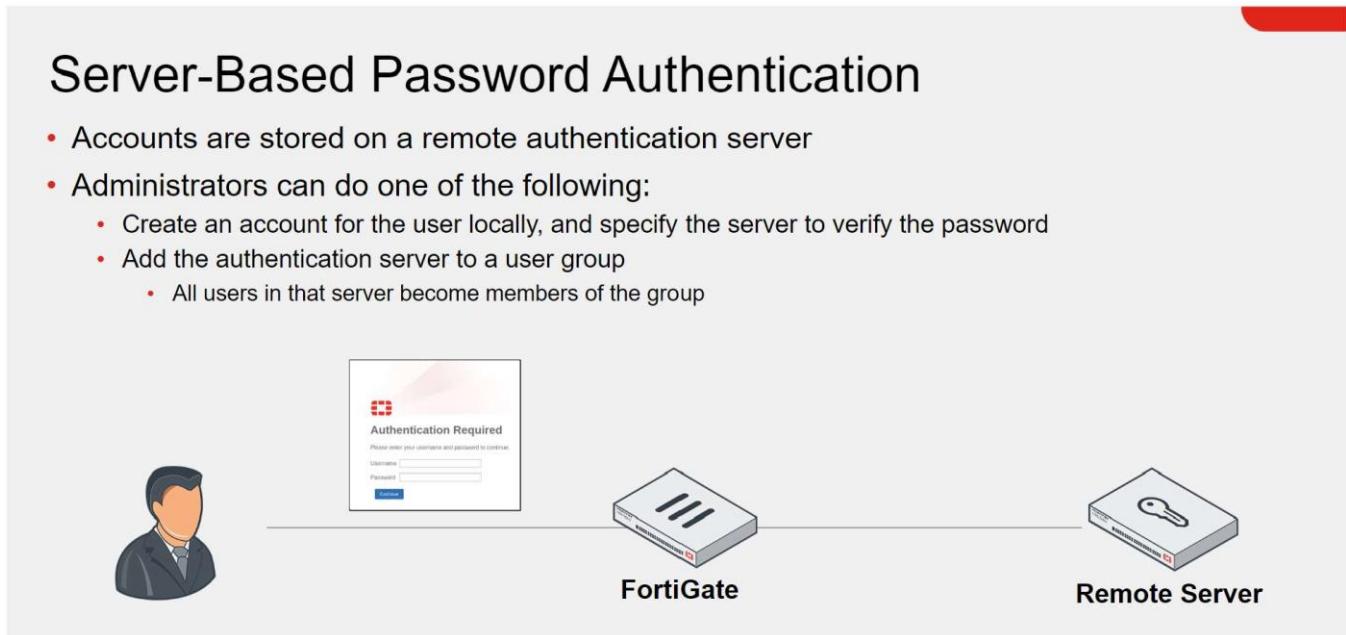
- Step 1: User Type** (highlighted with a green checkmark): Shows "Local User" selected. Other options include Remote RADIUS User, Remote TACACS+ User, Remote LDAP User, FSSO, and FortiIN.
- Step 2: Login Credentials**: Shows "Username: Student" and "Password: [REDACTED]" fields.
- Step 3: Contact Info**: Shows "Two-factor Authentication" checkbox.
- Step 4: Extra Info**: Shows "User Account Status: Enabled" and "User Group: [REDACTED]" fields.

**Bottom Navigation:** The left sidebar lists various FortiGate management categories, with "User & Authentication" and "User Definition" currently selected.

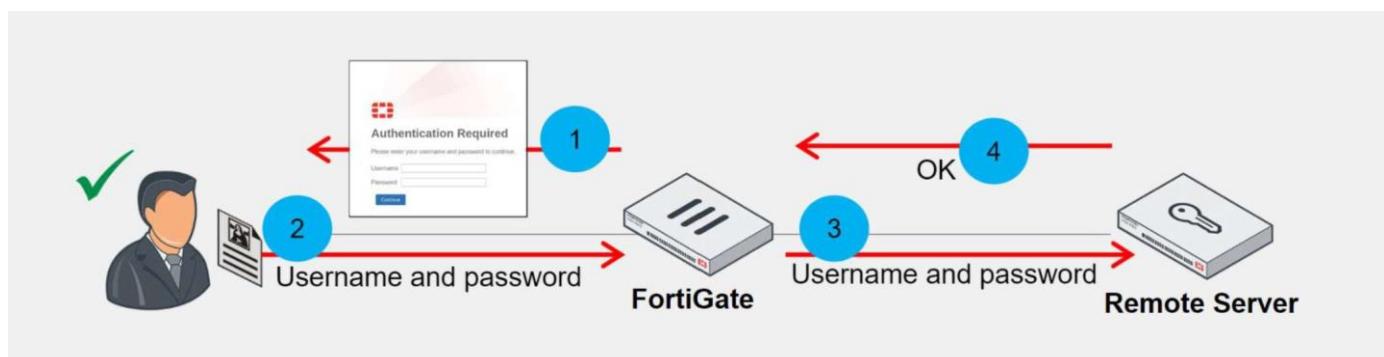
# Server-Based Password Authentication

When server-based password authentication is used, a **remote authentication server authenticates users**. This method is desirable when:

- Multiple FortiGate devices need to authenticate the same users or user groups,
- or when adding FortiGate to a network that already contains an authentication server.



When you use a remote authentication server to authenticate users, FortiGate sends the user's entered credentials to the remote authentication server. The remote authentication server responds by indicating whether the credentials are valid or not. If valid, FortiGate consults its configuration to deal with the traffic. Note that it is the remote authentication server—not FortiGate—that evaluates the user credentials.



When the server-based password authentication method is used, FortiGate does not store all (or, in the case of some configurations, any) of the user information locally.

# Server-Based Password Authentication – Users

## Server-Based Password Authentication—Users

- Create user accounts on FortiGate
  - Select remote server type and point to preconfigured remote server
  - Add user to a group
- Add the remote authentication server to user groups

Must be preconfigured on FortiGate

Edit User Group

Name: Remote-users

Type: Firewall

Members:

Remote Groups

+ Add Edit Delete

Remote Server: FortiAuth-RADIUS Group Name: Remote-users

OK Cancel

User & Authentication > User Definition

Users/Groups Creation Wizard

1 User Type > 2 RADIUS Server > 3 Contact Info > 4 Extra Info

Local User  
Remote RADIUS User  
Remote TACACS+ User  
Remote LDAP User  
FSSO  
FortiNAC User

Must be preconfigured on FortiGate

Users/Groups Creation Wizard

1 User Type > 2 RADIUS Server > 3 Contact Info > 4 Extra Info

Username: [redacted]

RADIUS Server: FortiAuth-RADIUS

You can configure FortiGate to use external authentication servers in the following two ways:

- **Create user accounts on FortiGate.** With this method, you must select the remote authentication server type (RADIUS, TACACS+, or LDAP), point FortiGate to your preconfigured remote authentication server, and add the user to an appropriate group. This is usually done when you want to add two-factor authentication to your remote users. Remember, POP3 is only configurable through the CLI.
- **Add the remote authentication server to user groups.** With this method, you must create a user group and add the preconfigured remote server to the group. Accordingly, any user who has an account on the remote authentication server can authenticate. If you are using other types of remote servers, such as an LDAP server, as the remote authentication server, you can control access to specific LDAP groups, as defined on the LDAP server.

Similar to local password authentication, you must then add the preconfigured user group (in which the user is a member) to a firewall policy in order to authenticate.

You will learn about user groups and firewall policies later in this lesson.

# LDAP Overview

---

LDAP is the **Lightweight Directory Access Protocol**. It's a standards-based protocol that sits on top of TCP/IP and allows clients to perform a variety of operations in a **directory server**, including: storing and retrieving data, searching for data matching a given set of criteria, authenticating clients, and more.

The standard TCP ports for LDAP are **389** for unencrypted communication and **636** for LDAP over an SSL/TLS-encrypted channel.

LDAP is a vendor-neutral software protocol used to lookup information or devices within a network. Whether you want to build a central authentication server for your organization or want to simplify access to internal servers and printers, LDAP is the answer.

## What is LDAP?

LDAP is a standard protocol designed to maintain and access “directory services” within a network. Think of a directory service as a phonebook for different network resources like files, printers, users, devices, and servers, etc.

For example, an organization may store information for all their printers in a directory. LDAP can enable users to search for a specific printer, locate it on the network, and securely connect to it.

**LDAP is widely used to build central authentication servers. These servers contain usernames and passwords for all the users within a network. Any-and-all applications and services can connect to the LDAP server to authenticate and authorize users.**

## What is LDAP authentication?

LDAP authentication is the process of verifying usernames and passwords stored in a directory service, like OpenLDAP or Microsoft Active Directory. Administrators can create user accounts within a directory and grant them permissions.

**When a user tries to access a resource, a request is sent to the LDAP authentication server. The LDAP server validates the entered username-password against the data in the directory. If there is a match, it then checks whether the user is authorized to access the requested resource.**

# LDAP vs Active Directory

LDAP and Active Directory are sometimes used interchangeably, but they are not the same thing. Active Directory is a proprietary directory service developed by Microsoft. It can be used for authentication, and/or storing information about network resources. LDAP is one of the protocols that is used to create or query objects in Active Directory.

**In a nutshell, LDAP is a language to talk to directory services, and Active Directory is one such directory service.**

## Directory Servers

A directory server (more technically referred to as a **Directory Server Agent**, a **Directory System Agent**, or a **DSA**) is a type of network database that stores information represented as trees of entries.

To understand what LDAP is, it's best to understand what it was used for in the first place: directory services.

Directory services securely manage users and their access rights to IT resources within an organization using certain protocols.

LDAP is one of the core protocols used for these services. Although directory services may use additional protocols like Kerberos, SAML, RADIUS, SMB, Oauth et.c., most still use LDAP today.

The following are some of the key elements and concepts of the LDAP protocol and LDAP-based directory:

## LDAP Directory Information Tree

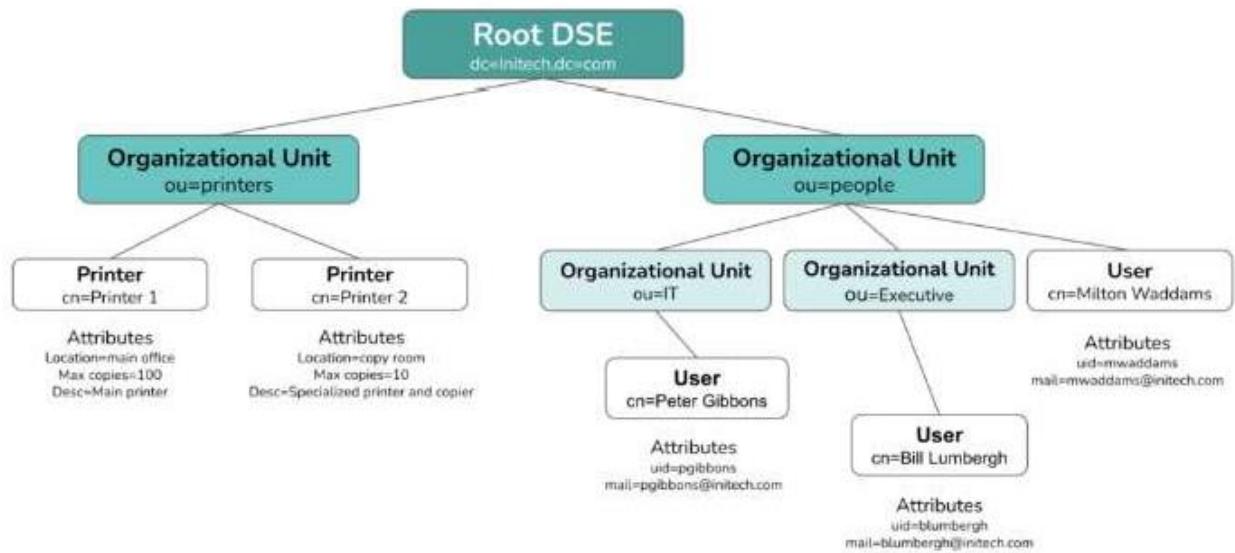
LDAP organizes information in a hierarchical tree structure called a directory information tree (DIT). The DIT can vary based on the software or directory service you use.

Generally, though, LDAP directories follow a tree structure where entries without subordinates — users, for example—are leaves. At the same time, the root is the overarching entity encompassing all the information within the directory.

**The root is a directory server agent specific entry, also called the root DSE, and it provides information about the directory.**

Below is a highly simplified example of an LDAP directory information tree (DIT). There can only be one root, but the branches can be iterative and groups can nest. The leaves—users and printers in the below diagram—have attributes, but they cannot have subordinate entities.

LDAP directories can contain entries for users, groups, printers, servers, applications, and more.



## Entries

An LDAP entry is a collection of information about an entity. Each entry consists of three primary components:

- a Distinguished Name (DN),
- a collection of Attributes,
- and a collection of Object Classes.

Entries use attributes to describe the real-world items stored in the directory, like a user or a machine. Just as you'll find in a phone book, or perhaps, your phone's contact list, users in a DIT exist as entries, which store additional information about the user.

In LDAP, entries are often referred to by their common name (CN). For users, this common name is usually their username or first and last name.

## Attribute

Attributes describe a user, server, or other item stored in the LDAP directory. A user's attributes, for starters, would typically include their full name, email address, username, and password.

Attributes are made up of a **type** and a **value**; i.e.,

```
mail(type)=pgibbons@initech.com(value)
```

## ObjectClass

The available attributes to include are predefined by an ObjectClass attribute. Organizations may use more than one ObjectClass attribute and create custom ObjectClass attributes to contain the information they want to store in their LDAP directory.

## DNs and RDNs

An entry's Distinguished Name, often referred to as a DN, uniquely identifies that entry and its position in the directory information tree (DIT) hierarchy. The DN of an LDAP entry is much like the path to a file on a filesystem.

An LDAP DN is comprised of zero or more elements called Relative Distinguished Names, or RDNs. Each RDN is comprised of one or more (usually just one) attribute-value pairs. For example, "uid=john.doe" represents an RDN comprised of an attribute named "uid" with a value of "john.doe". If an RDN has multiple attribute-value pairs, they are separated by plus signs, like "givenName=John+sn=Doe".

For DNs with multiple RDNs, the order of the RDNs specifies the position of the associated entry in the DIT. RDNs are separated by commas, and each RDN in a DN represents a level in the hierarchy in descending order (i.e., moving closer to the root of the tree, which is called the naming context). That is, if you remove an RDN from a DN, you get the DN of the entry considered the parent of the former DN. For example, the DN "uid=john.doe, ou=People, dc=example, dc=com" has four RDNs, with the parent DN being "ou=People, dc=example, dc=com".

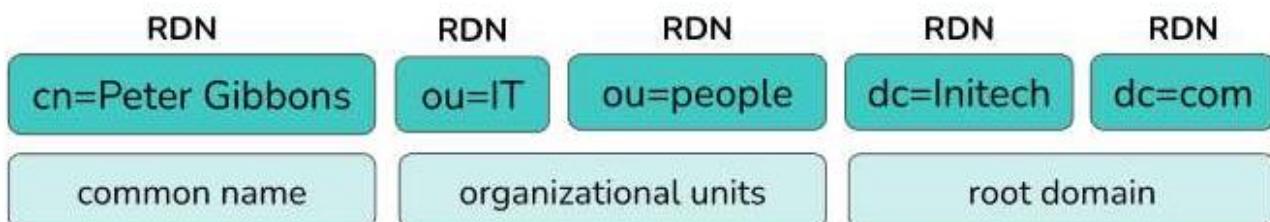
## DNs are formatted as follows:

RDN, RDN, RDN

For Peter Gibbons, a programmer in the IT department at Initech, the DN may look like this:

```
cn=Peter Gibbons,ou=IT,ou=People,dc=Initech,dc=com
```

**cn=Peter Gibbons,ou=IT,ou=People,dc=Initech,dc=com**



## Summary

LDAP protocol provides an interface with directories.

- An **entry** consists of a set of **attributes**.
- An attribute has a **name** (an *attribute type* or *attribute description*) and one or more **values**.
- Each entry has a unique identifier: its **Distinguished Name (DN)**. This consists of its **Relative Distinguished Name (RDN)**, constructed from some attribute(s) in the entry, followed by the parent entry's DN. Think of the DN as the full file path and the RDN as its relative filename in its parent folder (e.g. if **/foo/bar/myfile.txt** were the DN, then **myfile.txt** would be the RDN).

A DN may change over the lifetime of the entry, for instance, when entries are moved within a tree. To reliably and unambiguously identify entries, a UUID might be provided in the set of the entry's *operational attributes*.

An entry can look like this:

```
dn: cn=John Doe, dc=example, dc=com
cn: John Doe
givenName: John
sn: Doe
telephoneNumber: +1 888 555 6789
telephoneNumber: +1 888 555 1232
mail: john@example.com
manager: cn=Barbara Doe, dc=example, dc=com
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top
```

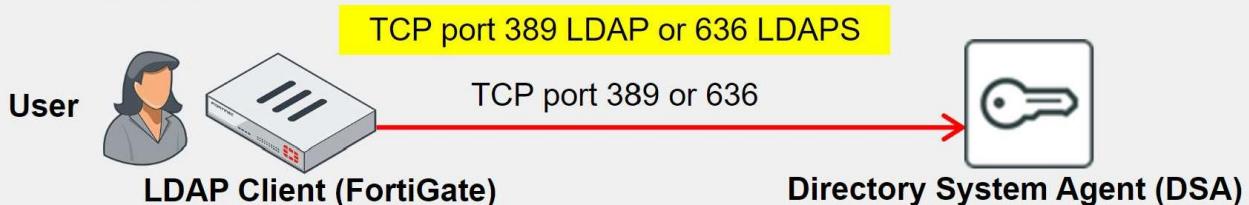
"**dn**" is the distinguished name of the entry; it is neither an attribute nor a part of the entry. "**cn=John Doe**" is the entry's RDN (Relative Distinguished Name), and "**dc=example, dc=com**" is the DN of the parent entry, where "**dc**" denotes '[Domain Component](#)'. The other lines show the attributes in the entry. Attribute names are typically mnemonic strings, like "**cn**" for common name, "**dc**" for domain component, "**mail**" for email address, and "**sn**" for surname.

A server holds a subtree starting from a specific entry, e.g., "**dc=example, dc=com**" and its children. Servers may also hold references to other servers, so an attempt to access "**ou=department, dc=example, dc=com**" could return a *referral* or *continuation reference* to a server that holds that part of the directory tree. The client can then contact the other server. Some servers also support *chaining*, which means the server contacts the other server and returns the results to the client.

**Lightweight Directory Access Protocol (LDAP)** is an application protocol used for accessing and maintaining distributed directory information services.

## LDAP Overview

- LDAP is an application protocol for accessing and maintaining distributed directory information services



- LDAP maintains authentication data, including:
  - Departments, people (and groups of people), passwords, email addresses, and printers
- LDAP consists of a data-representation scheme, a set of defined operations, and a request-and-response network
- Binding is the operation in which the LDAP server authenticates the user

The LDAP protocol is used to maintain authentication data that may include: **departments, people, groups of people, passwords, email addresses, and printers.**

LDAP consists of a **data-representation scheme, a set of defined operations, and a request-and-response network.**

The LDAP protocol includes a number of operations that a client can request, such as binding, search, compare, add, delete, and modify an entry. Binding is the operation in which the LDAP server authenticates the user. If the user is successfully authenticated, binding allows the user access to the LDAP server, based on that user's permissions.

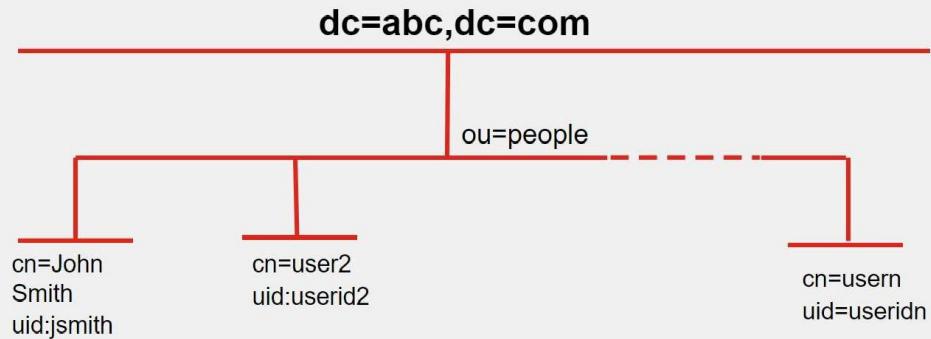
There are two types of LDAP protocol:

- **LDAP**
- **LDAPS**

Note that it is important to understand that **LDAP on port 389 is not secure** because it sends the password in clear text. **It is highly recommended to use LDAPS on port 636** which is more secure.

# LDAP Structure

## LDAP Structure



The LDAP structure is similar to a tree that contains entries (objects) in each branch. An LDAP server hierarchy often reflects the hierarchy of the organization it serves.

The root represents the organization itself, usually defined as domain component (DC), and a DNS domain, such as abc.com (because the name contains a dot, it is written as two parts separated by a comma: `dc=abc, dc=com`). You can add additional levels of hierarchy as needed, such as organizational unit (ou), user group (cn), user (uid) and so on.

The example shown on this slide is an LDAP hierarchy in which all user account entries reside at the organization unit (OU) level, just below DC.

**When requesting authentication, an LDAP client, such as a FortiGate device, must specify the part of the hierarchy where the user account record can be found. This is called the distinguished name (DN). In the example on this slide, DN is `ou=people, dc=abc, dc=com`.**

**The authentication request must also specify the particular user account entry.** Although this is often called the **common name (CN)**, the identifier you use is not necessarily CN. On a computer network, it is appropriate to **use UID**, the person's user ID, because that is the information that they will provide when they log in.

# Configuring an LDAP Server on FortiGate

## Configuring an LDAP Server on FortiGate

Directory tree attribute that identifies users

Part of the hierarchy where user records exist

Credentials for an LDAP administrator

On the **LDAP Servers** page, you can configure FortiGate to point to an LDAP server for server-based password authentication. The configuration depends heavily on the server's schema and security settings. Windows Active Directory (AD) is very common.

The **Common Name Identifier** setting is the attribute name you use to find the username. Some schemas allow you to use the attribute **userid (uid)**. AD most commonly uses **sAMAccountName** or **cn**, but can use others as well.

The **Distinguished Name** setting identifies the top of the tree where the users are located, which is generally the dc value; however, it can be a specific container or OU. You must use the correct X.500 or LDAP format.

The **Bind Type** setting depends on the security settings of the LDAP server. You must use the setting **Regular** (to specify a regular bind) if you are searching across multiple domains and require the credentials of a user that is authorized to perform LDAP queries (for example, an LDAP administrator).

If you want to have a **secure connection** between FortiGate and the remote LDAP server, enable **Secure Connection** and include the LDAP server protocol (LDAPS or STARTTLS) as well as the CA certificate that verifies the server certificate. LDAPS uses port 636 for communication.

The **Test Connectivity** button tests only whether the connection to the LDAP server is successful or not. To test whether a user's credentials can successfully authenticate, you can use the **Test User Credentials** button or use the **CLI**.

## To configure an LDAP server on the FortiGate:

1. Go to **User & Authentication > LDAP Servers**.
2. Click **Create New**.
3. Configure the following:

Name	This connection name is for reference within the FortiGate only.
Server IP/Name	LDAP server IP address or FQDN resolvable by the FortiGate.
Server Port	By default, LDAP uses port 389 and LDAPS uses 636. Use this field to specify a custom port if necessary.
Common Name Identifier	Attribute field of the object in LDAP that the FortiGate uses to identify the connecting user. The identifier is case sensitive. Common attributes are: <ul style="list-style-type: none"><li>• <i>cn</i> (Common Name)</li><li>• <i>sAMAccountName</i> (SAMAccountName)</li><li>• <i>uid</i> (User ID)</li></ul>
Distinguished Name	Used to look up user account entries on the LDAP server. It reflects the hierarchy of LDAP database object classes above the CN identifier in which you are doing the lookup. Enter <i>dc=COMPANY,dc=com</i> to specify the root of the domain to include all objects. Enter <i>ou=VPN-Users,dc=COMPANY,dc=com</i> to look up users under a specific organization unit.
Exchange server	Enable to specify the exchange server connector to collect information about authenticated users from a corporate exchange server. See <a href="#">Exchange Server connector</a> for more details.
Bind Type	Select one of the following options: <ul style="list-style-type: none"><li>• <i>Simple</i>: bind using simple password authentication using the client name. The LDAP server only looks up against the distinguished name (DN), but does not search on the subtree.</li><li>• <i>Anonymous</i>: bind using an anonymous user, and search starting from the DN and recurse over the subtrees. Many LDAP servers do not allow this.</li><li>• <i>Regular</i>: bind using the username and password provided, and search starting from the DN and recurse over the subtrees.</li></ul>
Username	If using regular bind, enter a username with sufficient privileges to access the LDAP server. The following formats are supported: <ul style="list-style-type: none"><li>• <i>username\administrator</i></li><li>• <i>administrator@domain</i></li><li>• <i>cn=administrator,cn=users,dc=domain,dc=com</i></li></ul>
Password	If using regular bind, enter the password associated with the username.
Secure Connection	Enable to apply security to the LDAP connection through STARTTLS or LDAPS.
Protocol	If <i>Secure Connection</i> is enabled, select <i>STARTTLS</i> or <i>LDAPS</i> . Selecting <i>STARTTLS</i> changes the port to 389 and selecting <i>LDAPS</i> changes the port to 636.
Certificate	Enable and select the certificate so the FortiGate will only accept a certificate from the LDAP server that is signed by this CA.
Server identity check	Enable to verify the server domain or IP address against the server certificate. This option is enabled by default and it is recommended to leave it enabled for a secure configuration.

4. Optionally, click ***Test User Credentials*** to ensure that the account has sufficient access rights.
5. Click OK.

The FortiGate checks the connection and updates the Connection Status.

## To configure a secure connection to the LDAP server in the GUI:

1. Go to **User & Authentication > LDAP Servers**.
2. Click **Create New**.
3. Configure the following:

Name	LDAP-fortiad
Server IP/Name	10.88.0.1
Server Port	636
Common Name Identifier	sAMAccountName
Distinguished Name	dc=fortiad,dc=info
Exchange server	Disabled
Bind Type	Regular Enter the <i>Username</i> and <i>Password</i> for LDAP binding and lookup.
Secure Connection	Enabled <ul style="list-style-type: none"> <li>• Set <i>Protocol</i> to <i>LDAPS</i>.</li> <li>• Enable <i>Certificate</i> and select the CA certificate to validate the server certificate.</li> </ul>
Server identity check	Optionally, enable to verify the domain name or IP address against the server certificate.

Edit LDAP Server

Name	LDAP-fortiad
Server IP/Name	10.88.0.1
Server Port	636
Common Name Identifier	sAMAccountName
Distinguished Name	dc=fortiad,dc=info
Exchange server	<input checked="" type="checkbox"/>
Bind Type	Simple   Anonymous   <b>Regular</b>
Username	fortiad\Administrator
Password	*****
Secure Connection	<input checked="" type="checkbox"/>
Protocol	STARTTLS   <b>LDAPS</b>
Certificate	<input checked="" type="checkbox"/> CA_Cert_1
Server identity check	<input checked="" type="checkbox"/> <small>i</small>
Connection status	<input checked="" type="checkbox"/> Successful
<a href="#">Test Connectivity</a>	
<a href="#">Test User Credentials</a>	

FortiGate

 [FortiGate-VM64-KVM](#)

Additional Information

[API Preview](#)

[References](#)

[Edit in CLI](#)

[Online Guides](#)

 [Relevant Documentation](#)

 [Video Tutorials](#)

 [Hot Questions at FortiAnswers](#)

 [Join the Discussion](#)

**OK** **Cancel**

1. Click **Test Connectivity** to verify the connection to the server.

5. Click **OK**.

# RADIUS Overview

RADIUS is much different from LDAP, because **there is no directory tree structure** to consider. RADIUS is a standard protocol that provides Authentication, Authorization, and Accounting (AAA) services.

## RADIUS Overview

- RADIUS is a standard protocol that provides AAA services



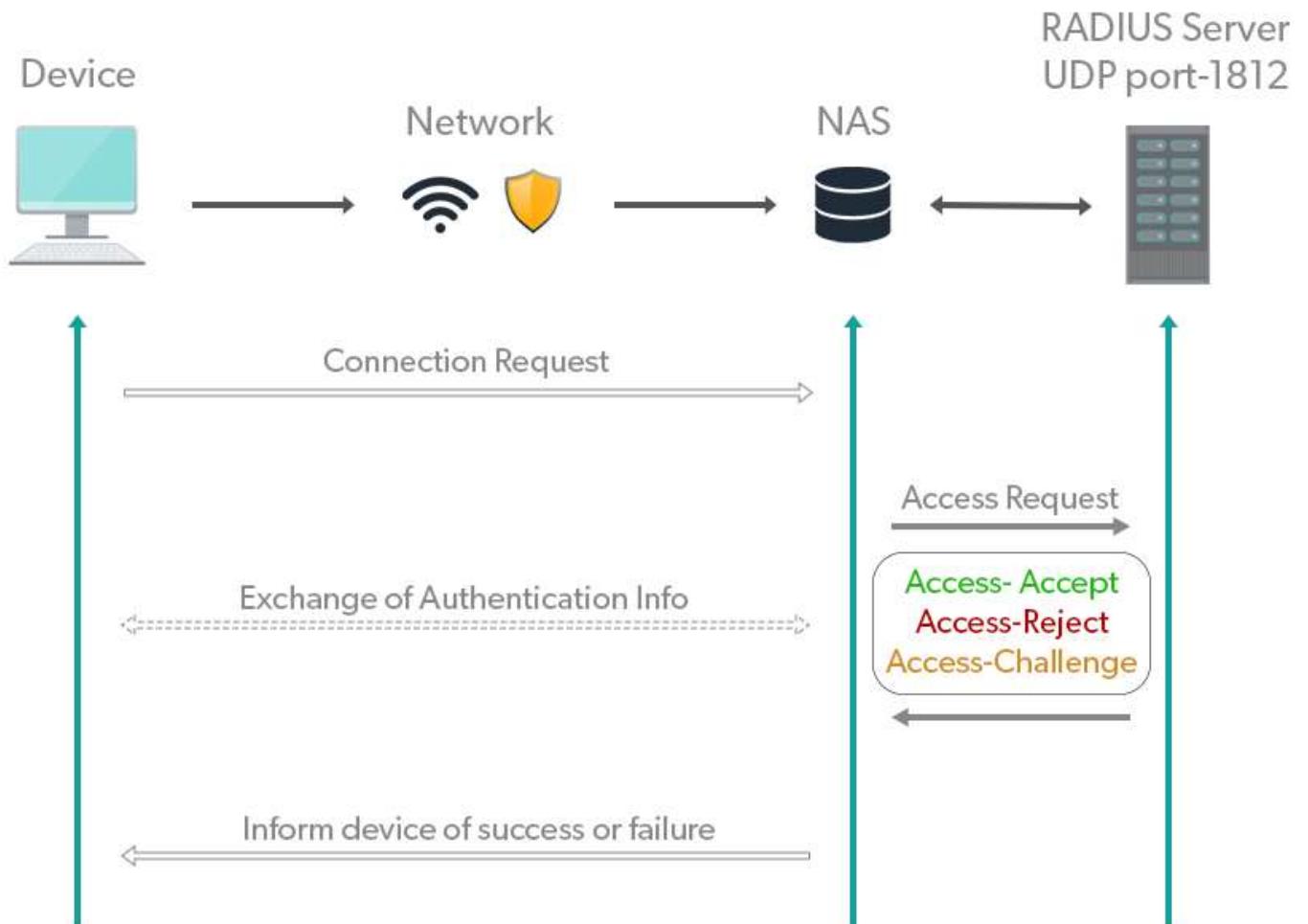
When a user is authenticating, the client (FortiGate) sends an **ACCESS-REQUEST** packet to the RADIUS server. The reply from the server is one of the following:

- **ACCESS-ACCEPT**, which means that the user credentials are correct
- **ACCESS-REJECT**, which means that the credentials are wrong
- **ACCESS-CHALLENGE**, which means that the server is requesting a secondary password ID, token, or certificate. This is typically the reply from the server **when using two-factor authentication**.

Not all RADIUS clients support the RADIUS challenge method.

RADIUS is an open-standard AAA protocol that uses **UDP port 1645 or 1812 for authentication** and **UDP port 1646 or 1813 for accounting**.

## RADIUS Authentication Process



# Configuring a RADIUS Server on FortiGate

A RADIUS server can be configured in the GUI by going to **User & Authentication > RADIUS Servers**, or in the **CLI** under config user radius.

The screenshot shows the 'User & Authentication > RADIUS Servers' configuration page. It displays fields for a new RADIUS server named 'FortiAuth-RADIUS' using the 'Default' authentication method. The 'Primary Server' section shows an IP/Name of '10.0.1.150' and a secret consisting of five asterisks ('\*\*\*\*\*'). A red box highlights the 'Test Connectivity' button. Two blue callout boxes provide instructions: one for entering the RADIUS server's IP/FQDN and another for entering its secret.

You can configure FortiGate to point to a RADIUS server for server-based password authentication through the **RADIUS Servers** page.

The **Primary Server IP/Name** setting is the IP address or FQDN of the RADIUS server.

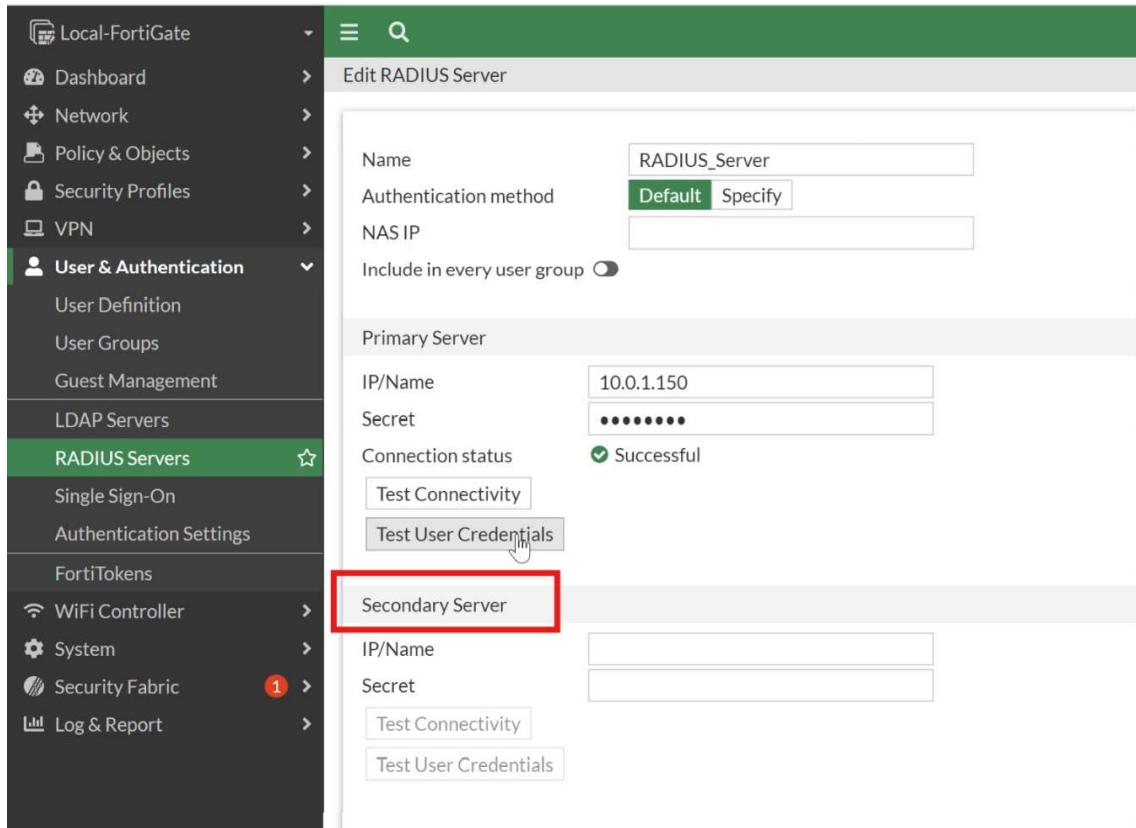
The **Primary Server Secret** setting is the secret that was set up on the RADIUS server in order to allow remote queries from this client. Backup servers (with separate secrets) can be defined in case the primary server fails. Note that FortiGate must be listed on the RADIUS server as a client of that RADIUS server or else the server will not reply to queries done by FortiGate.

The **Authentication Method** setting refers to the authentication protocol that the RADIUS server supports. Options include **chap**, **pap**, **mschap**, and **mschap2**. If you select Default, FortiGate will use pap, mschap2, and chap (in that order).

The **Test Connectivity** button tests only whether the connection to the RADIUS server is successful or not. To test whether a user's credentials can successfully authenticate, you can use the **Test User Credentials** button or the CLI.

The **Include in every User Group** option adds the RADIUS server and all users who can authenticate against it, to every user group created on FortiGate. So, you should enable this option only in very specific scenarios (for example, when only administrators can authenticate against the RADIUS server and policies are ordered from least restrictive to most restrictive).

GUI field	CLI setting	Description
Name	edit <name>	Define the RADIUS server object within FortiOS.
Authentication method	set auth-type {auto   ms_chap_v2   ms_chap   chap   pap}	Specify the authentication method, or select Default/auto to negotiate PAP, MSCHAP_v2, and CHAP in that order.
NAS IP	set nas-ip <IPv4_address>	Optional setting, also known as Calling-Station-Id. Specify the IP address the FortiGate uses to communicate with the RADIUS server. If left unconfigured, the FortiGate will use the IP address of the interface that communicates with the RADIUS server.
Include in every user group	set all-usergroup {enable   disable}	Optional setting to add the RADIUS server to each user group. This allows each user group to try and authenticate users against the RADIUS server if local authentication fails.
<b>Primary Server</b>		
IP/Name	set server <string>	Enter the IP address or resolvable FQDN of the RADIUS server.
Secret	set secret <password>	Enter the password used to connect to the RADIUS server.



# Testing the LDAP & RADIUS Query on the CLI

## Testing the LDAP and RADIUS Query on the CLI

- diagnose test authserver ldap <server\_name> <username> <password>
- Example:

```
# diagnose test authserver ldap External_Server aduser1 Training!  
authenticate 'aduser1' against 'External_Server' succeeded!  
Group membership(s) - CN=AD-users,OU=Training,DC=trainingAD,DC=training,DC=lab
```

- diagnose test authserver radius <server\_name> <scheme> <user> <password>

- Example:

```
# diagnose test authserver radius FortiAuth-RADIUS pap student fortinet  
authenticate 'student' against 'pap' succeeded, server=primary  
assigned_rad_session_id=810153440 session timeout=0 secs!  
Group membership(s) - remote-RADIUS-admins
```

Group memberships are provided by vendor-specific attributes configured on the RADIUS server

Use the `diagnose test authserver` command on the CLI to test whether a user's credentials can successfully authenticate. **You want to ensure that authentication is successful, before implementing it on any of your firewall policies.**

The response from the server reports success, failure, and group membership details.

Testing RADIUS is much the same as testing LDAP. Use the `diagnose test authserver` command on the CLI to test whether a user's credentials can successfully authenticate. Again, you should do this to ensure authentication is successful before implementing it on any of your firewall policies.

Like LDAP, it reports success, failure, and group membership details, depending on the server's response. Deeper troubleshooting usually requires RADIUS server access.

Note that Fortinet has a **vendor-specific attributes (VSA) dictionary to identify the Fortinet-proprietary RADIUS attributes**. This capability allows you to extend the basic functionality of RADIUS.

# Two-Factor Authentication

## Two-Factor Authentication

- Strong authentication that improves security by preventing attacks associated with the use of static passwords alone
- Requires two independent methods of identifying a user:
  - Something you know, such as a password or PIN
  - Something you have, such as a token or certificate

**Traditional user authentication** requires **your username** plus **something you know**, such as a **password**. The weakness in this traditional method of authentication is that if someone obtains your username, they need only your password to compromise your account. Furthermore, since people tend to use the same password across multiple accounts (some sites having more security vulnerabilities than others), accounts are vulnerable to attack, regardless of password strength.

**Two-factor authentication**, on the other hand, requires **something you know**, such as a **password**, and **something you have**, such as a **token** or **certificate**. Because this method places less importance on often vulnerable passwords, it makes compromising the account more complex for an attacker.

You can use two-factor authentication on FortiGate with both **user** and **administrator** accounts. The user (or user group to which the user belongs) is added to a firewall policy in order to authenticate.

Note that you cannot use two-factor authentication with explicit proxies.

## Two-Factor Authentication (Contd)

- One-time passwords (OTPs) can be used one time only
  - OTPs are more secure than static passwords
- Available on both user and administrator accounts
  - The user or user group is added to a firewall policy in order to authenticate
- Methods of OTP delivery include:
  - FortiToken 200 or FortiToken Mobile
    - Generates a six-digit code every 60 seconds based on a unique seed and GMT time
  - Email or SMS
    - An OTP is sent to the user's email or SMS
    - Email or SMS must be configured on the user's account
  - FortiToken mobile push
    - Supports two-factor authentication without requiring user to enter code
- NTP server recommended!

You can use **One-Time Passwords (OTPs)** as your second factor. OTPs are more secure than static passwords because **the passcode changes at regular intervals** and is valid for only a short amount of time.

Once you use the OTP, you can't use it again. So, even if it is intercepted, it is useless.

### Methods of OTP delivery:

FortiGate can deliver OTPs through tokens, such as **FortiToken 200** (hardware token) and **FortiToken Mobile** (software token), as well as through **email or SMS**. To deliver an OTP over email or SMS, the user account must contain user contact information.

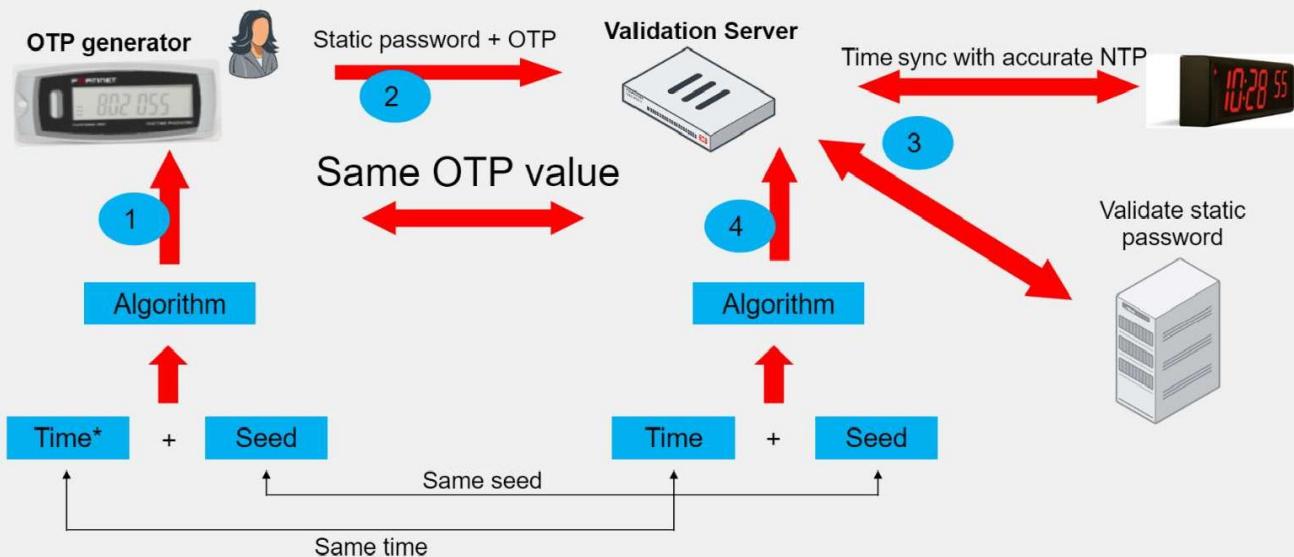
FortiTokens and OTPs delivered through email and SMS are time based. FortiTokens, for example, generate a new, six-digit password every 60 seconds (by default).

An NTP server is highly recommended to ensure the OTPs remain in sync.

FortiToken Mobile Push allows users to accept the authorization request from their FortiToken mobile app, without the need to enter an additional code.

# FortiTokens

## FortiTokens



Tokens use a **specific algorithm** to generate an OTP. The algorithm consists of:

- A **seed**: a unique, randomly-generated number that does not change over time
- The **time**: obtained from an accurate internal clock

Both seed and time go through an algorithm that generates an OTP (or passcode) on the token. The passcode has a short life span, usually measured in seconds (60 seconds for FortiToken 200, possibly more or less for other RSA key generators). Once the life span ends, a new passcode generates.

When using two-factor authentication using a token, the user must first log in with a static password followed by the passcode generated by the token. A validation server (FortiGate) receives the user's credentials and validates the static password first. The validation server then proceeds to validate the passcode. It does so by regenerating the same passcode using the seed and system time (which is synchronized with the one on the token) and comparing it with the one received from the user. If the static password is valid, and the OTP matches, the user is successfully authenticated. Again, both the token and the validation server must use the same seed and have synchronized system clocks. As such, it is crucial that you configure the date and time correctly on FortiGate, or link it to an NTP server (which is recommended).

# Assigning a FortiToken to a User

## Assigning a FortiToken to a User

User & Authentication > FortiTokens

Type	Serial Number	Status	User	Drift	Comments
Mobile Token	FTKMOB6B91B33BE5	Available	0	0	
Mobile Token	FTKMOB68BCB3CCB31	Available	0	0	

Two free FortiToken Mobile activations

New FortiToken

Type: Hard Token

Comments: Write a comment... 0/255

Serial Number:

Import:

New FortiToken

Type: Hard Token

Activation Code: 0000-0000-0000-0000-0000

Username: student

User Account Status: Enabled

User Type: Local User

User Group: Remote-users

Two-factor Authentication:  Two-factor Authentication

Authentication Type: FortiToken Cloud

Token: FTKMOB6B91B33BE5

Email Address:

SMS:

- Enable **Two-factor Authentication** and select the registered FortiToken

Can add a user to a group and create a firewall policy based on the user group

You can add a FortiToken 200 or FortiToken Mobile to FortiGate on the **FortiTokens** page.

A **Hard Token** has a serial number that provides FortiGate with details on the initial seed value. If you have several hard tokens to add, you can import a text file, where one serial number is listed per line.

A **Soft Token** requires an activation code. Note that each FortiGate (and FortiGate VM) provides two free FortiToken Mobile activations. You must purchase any additional tokens from Fortinet.

You cannot register the same FortiToken on more than one FortiGate. If you want to use the same FortiToken for authentication on multiple FortiGate devices, you must use a central validation server, such as **FortiAuthenticator**. In that case, FortiTokens are registered and assigned to users on FortiAuthenticator, and FortiGate uses FortiAuthenticator as its validation server.

After you have registered the FortiToken devices with FortiGate, you can assign them to users to use as their second-factor authentication method. To assign a token, edit (or create) the user account and select Enable Two-factor Authentication. In the Token field, select the registered token you want to assign.

# FortiToken in detail

FortiTokens are security tokens used as part of a Multi-Factor Authentication (MFA) system on FortiGate and FortiAuthenticator. A security token is a 6-digit or 8-digit (configurable) One-Time Password (OTP) that is used to authenticate one's identity electronically as a prerequisite for accessing network resources. FortiToken is available as either a mobile or a physical (hard) token. Mobile tokens can be purchased as a license, or consumed with points as part of the FortiToken Cloud service.

FortiToken Mobile and physical FortiTokens store their encryption seeds on the cloud. FortiToken Mobile seeds are generated dynamically when the token is provisioned. They are always encrypted whether in motion or at rest.

You can only register FortiTokens to a single FortiGate or FortiAuthenticator for security purposes. This prevents malicious third parties from making fraudulent requests to hijack your FortiTokens by registering them on another FortiGate or FortiAuthenticator.

Common usage for FortiTokens includes:

- Applying MFA to a VPN dialup user connecting to the corporate network
- Applying MFA to FortiGate administrators
- Applying MFA to firewall authentication and captive portal authentication

---

The MFA process commonly involves:

- **Something you know:** User password
- **Something you have:** The FortiToken OTP



A third factor of authentication is added to the authentication process:

- **Something you are:** Your fingerprint or face

To enable the third factor, refer to the [Activating FortiToken Mobile on a mobile phone](#)

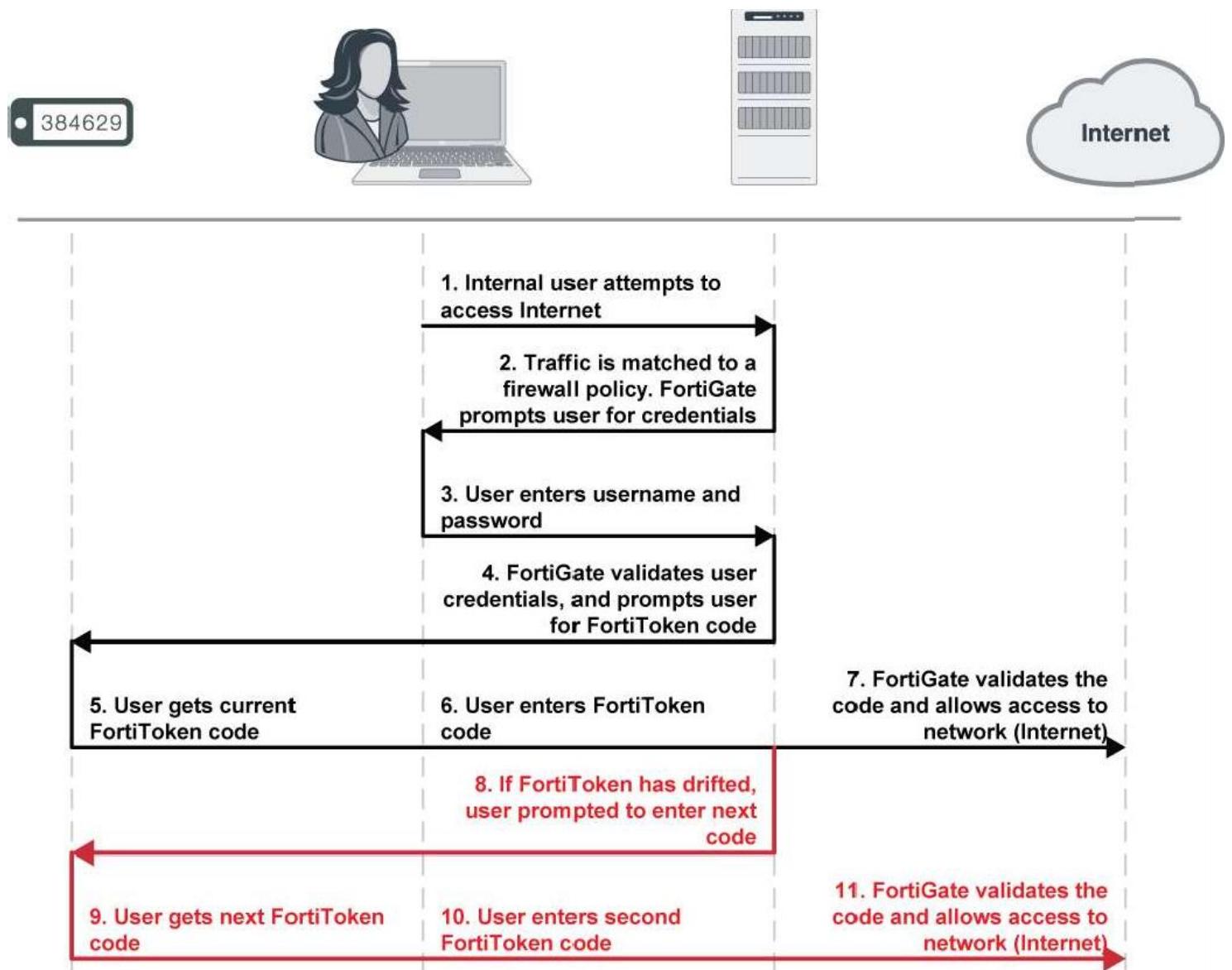
---

## The following illustrates the FortiToken MFA process:

1. The user attempts to access a network resource.
2. FortiOS matches the traffic to an authentication security policy and prompts the user for their username and password.
3. The user enters their username and password.
4. FortiOS verifies their credentials. If valid, it prompts the user for the FortiToken code.
5. The user views the current code on their FortiToken. They enter the code at the prompt.
6. FortiOS verifies the FortiToken code. If valid, it allows the user access to network resources.

**If the FortiToken has drifted, the following must take place for the FortiToken to resynchronize with FortiOS:**

1. FortiOS prompts the user to enter a second code to confirm.
2. The user gets the next code from the FortiToken. They enter the code at the prompt.
3. FortiOS uses both codes to update its clock to match the FortiToken.



# FortiToken Mobile

FortiToken Mobile is an OATH compliant, event- and time-based one-time password (OTP) generator for mobile devices. It provides an easy and flexible way to deploy and provision FortiTokens to your end users through mobile devices. FortiToken Mobile produces its OTP codes in an application that you can download onto your Android or iOS mobile device without the need for a physical token.

You can download the free FortiToken Mobile application for Android from the [Google Play Store](#), and for iOS from the [Apple App Store](#).

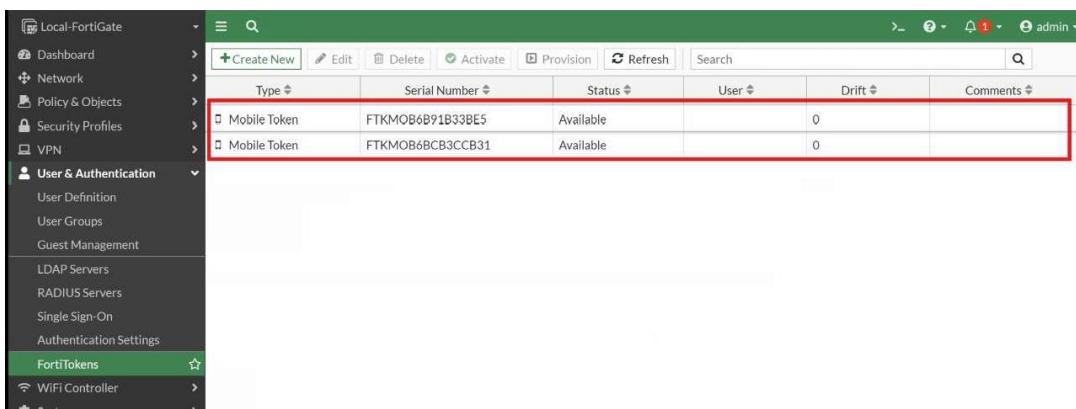
## Registering FortiToken Mobile

To deploy FortiToken Mobile for your end users, you must first register the tokens on your FortiGate. After registering the tokens, you can assign them to your end users.

Each FortiGate comes with two free FortiToken Mobile tokens. These tokens should appear under *User & Authentication > FortiTokens*. If no tokens appear, you may import them. Ensure that your FortiGate is registered and has internet access to connect to the FortiToken servers to import the tokens.

### To import FortiTokens from the FortiGate GUI:

1. Go to *User & Authentication > FortiTokens*.
2. Click the *Import Free Trial Tokens* icon at the top. The two free tokens are imported.



The screenshot shows the FortiGate User & Authentication interface with the FortiTokens tab selected. A red box highlights the table containing the two imported tokens:

Type	Serial Number	Status	User	Drift	Comments
Mobile Token	FTKMOB6B91B33BE5	Available		0	
Mobile Token	FTKMOB6BCB3CCB31	Available		0	



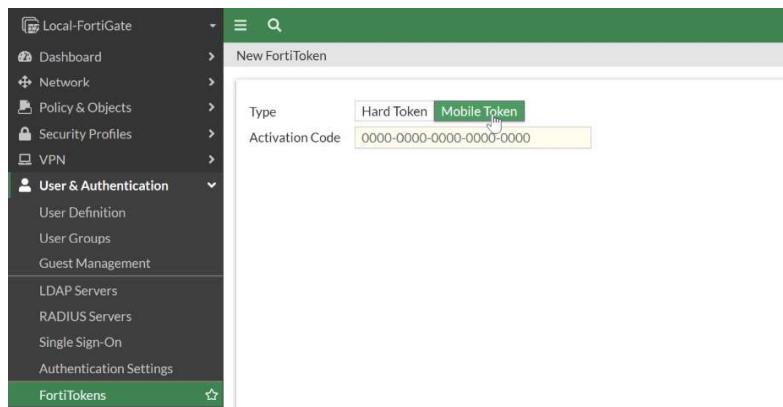
If only one free token appears, you can first delete that token and then follow the procedure to import the two free tokens from either the GUI or the CLI.

---

If you have the [FortiToken Mobile redemption certificate](#), you can register FortiToken Mobile on a FortiGate.

## To register FortiToken Mobile from the FortiGate GUI:

1. Go to *User & Authentication* > *FortiTokens* and click *Create New*. The New FortiToken dialog appears.
2. For the *Type* field, select *Mobile Token*.
3. Locate the 20-digit code on the redemption certificate and type it in the *Activation Code* field.
4. Click *OK*. The token is successfully registered.



 If you attempt to add invalid FortiToken serial numbers, there is no error message. FortiOS does not add invalid serial numbers to the list.

## Provisioning FortiToken Mobile

Once registered, FortiTokens need to be provisioned for users before they can be activated. In this example, you will provision a mobile token for a local user. Similar steps can be taken to assign FortiTokens to other types of users.

## To create a local user and assign a FortiToken in the FortiGate GUI:

1. Go to *User & Authentication* > *User Definition*, and click *Create New*. The *Users/Groups Creation Wizard* appears.
2. In the *User Type* tab, select *Local User*, and click *Next*.

## Users/Groups Creation Wizard

① User Type > ② Login Credentials > ③ Contact Info > ④ Extra Info

Local User

- Remote RADIUS User
- Remote TACACS+ User
- Remote LDAP User
- FSSO
- FortiClient EMS User
- FortiNAC User

< Back

Next

Cancel

3. In the *Login Credentials* tab, enter a *Username* and *Password* for the user, and click *Next*.

## Users/Groups Creation Wizard

① User Type > ② Login Credentials > ③ Contact Info > ④ Extra Info

Username

Password

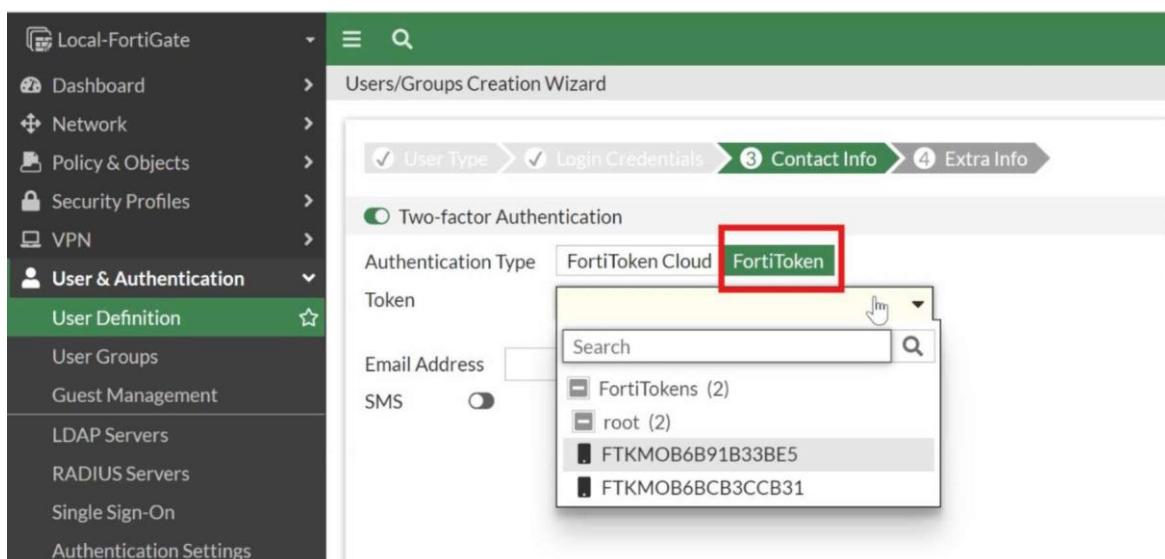
< Back

Next

Cancel

4. In the *Contact Info* tab:

1. Enable the *Two-factor Authentication* toggle.
2. Select *FortiToken* for *Authentication Type*.
3. Select a Token to assign to the user from the drop-down list.
4. Enter the user's email address in the *Email Address* field. This is the email where the user will receive the QR code for activation of the FortiToken.
5. Click *Next*.



5. In the *Extra Info* tab, make sure the *User Account Status* field is set to Enabled. You can also optionally assign the user to a user group by enabling the *User Group* toggle.

### Users/Groups Creation Wizard

4 User Type > 4 Login Credentials > 4 Contact Info > 4 Extra Info

User Account Status:  Enabled  Disabled

User Group:

< Back  Submit  Cancel

6. Click *Submit*. An activation code should be sent to the created user by email or SMS, depending upon the delivery method configured above.



FortiGate has the *Email Service* setting configured using the server *notifications.fortinet.net* by default. To see configuration, go to *System > Settings > Email Service*.

---

The activation code expires if not activated within the 3-day time period by default. However, the expiry time period is configurable.

# Activating FortiToken Mobile on a mobile phone

After your system administrator provisions your token, you receive a notification with an activation code and expiry date via SMS or email. If you do not activate your token by the expiry date, you must contact your system administrator so that they can reassign your token for activation.

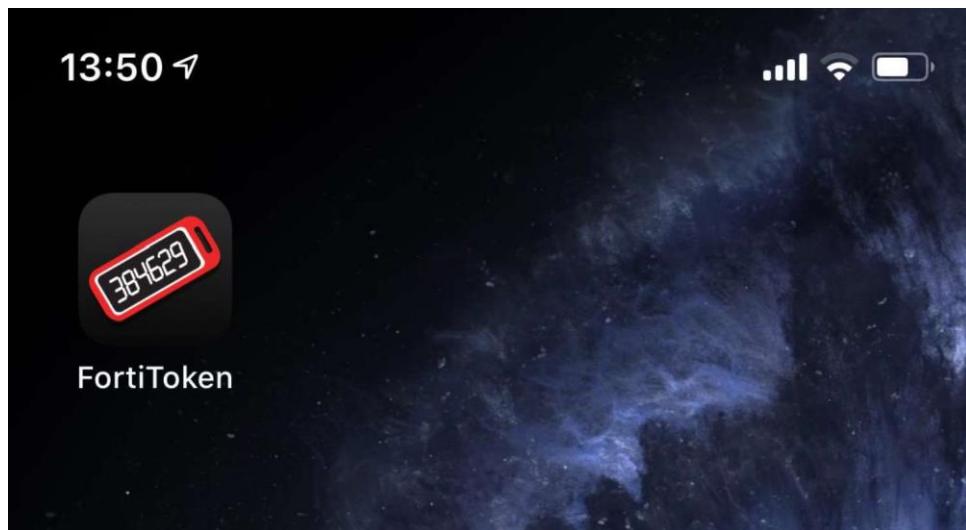
Platforms that support FortiToken Mobile:

Platform	Device and firmware support
iOS	iPhone, iPad, and iPod Touch with iOS 6.0 and later.
Android	Phones and tablets with Android Jellybean 4.1 and later.
Windows	Windows 10 (desktop and mobile), Windows Phone 8.1, and Windows Phone 8

The following instructions describe procedures when using FortiToken Mobile for iOS on an iPhone. Procedures may vary depending on your device and firmware.

## To activate FortiToken Mobile on iOS:

1. On your iOS device, tap on the FortiToken application icon to open the application. If this is your first time opening the application, it may prompt you to create a PIN for secure access to the application and tokens.



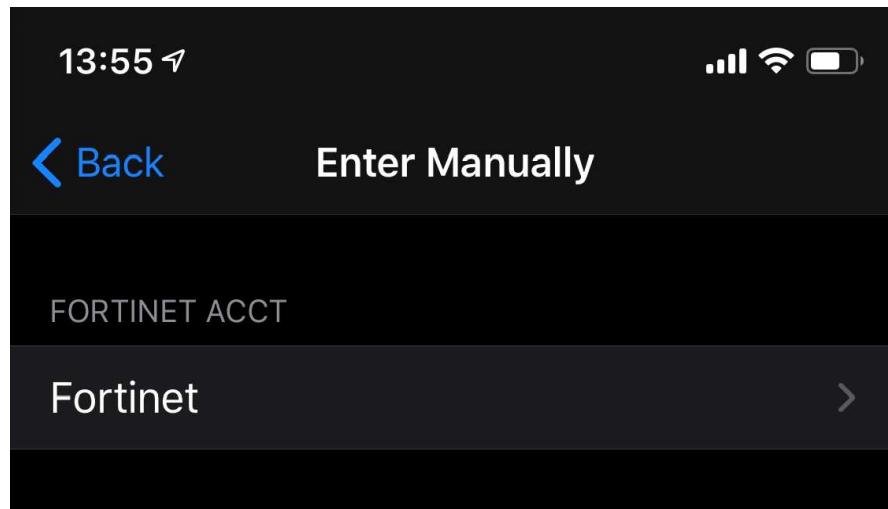
2. Tap on the + icon. The *Scan Barcode* screen appears.



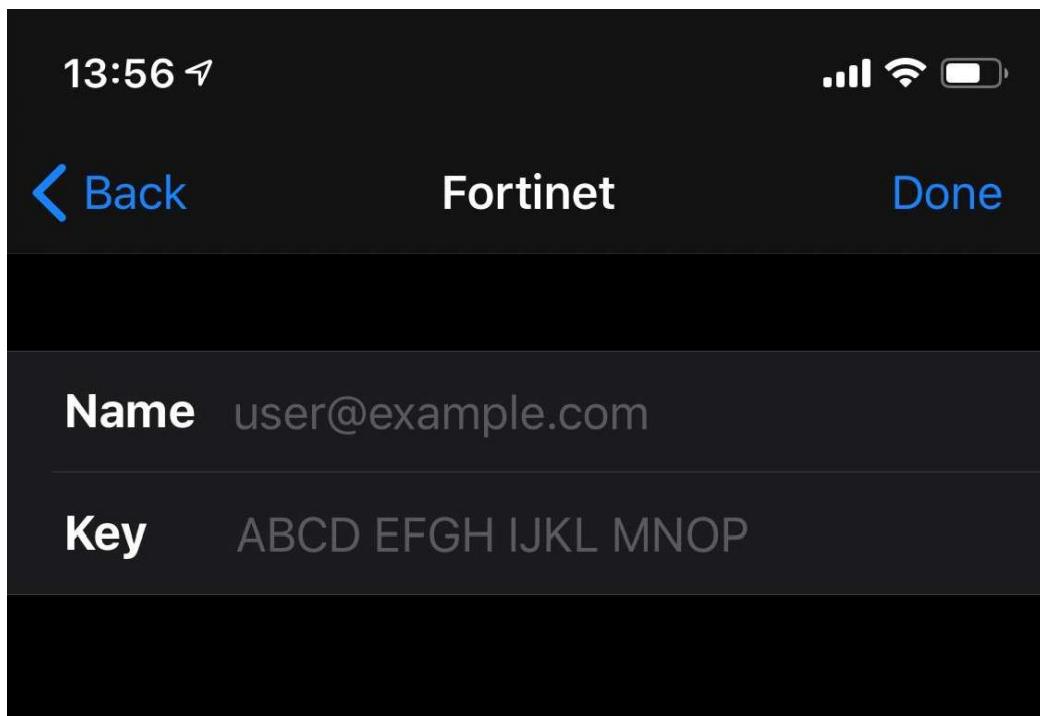
3. If you received the QR code via email, locate and scan the QR code in your email.

**OR**

If you received the activation key via SMS, tap on Enter Manually at the bottom of the screen, and tap on *Fortinet*.



Enter your email address in the Name field, the activation key in the Key field, and tap Done.



4. FortiToken Mobile activates your token, and starts generating OTP digits immediately. To view or hide the OTP digits, tap the eye icon.



After you open the application, FortiToken Mobile generates a new 6-digit OTP every 30 seconds. All configured tokens display on the application homescreen.

The FortiToken Mobile activation process described above caters to the MFA process that involves two factors (password and OTP) of the authentication process.

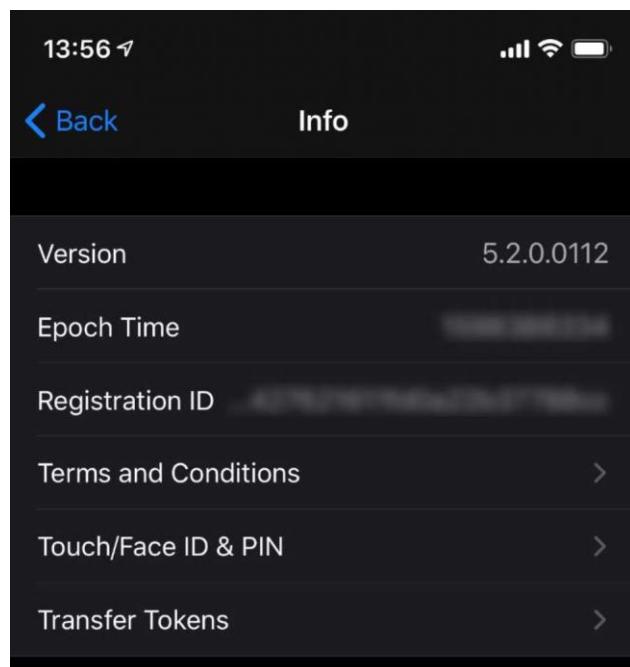
A third factor (fingerprint or face) can be enabled as well.

### To enable **Touch/Face ID** on iOS for FortiToken Mobile:

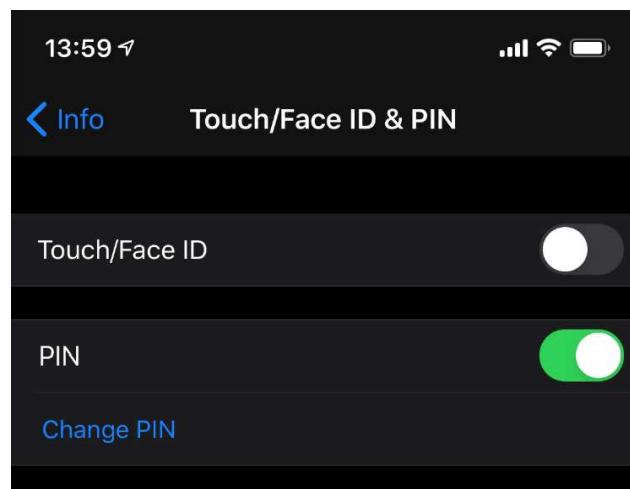
1. Open the FortiToken application and tap on Info.



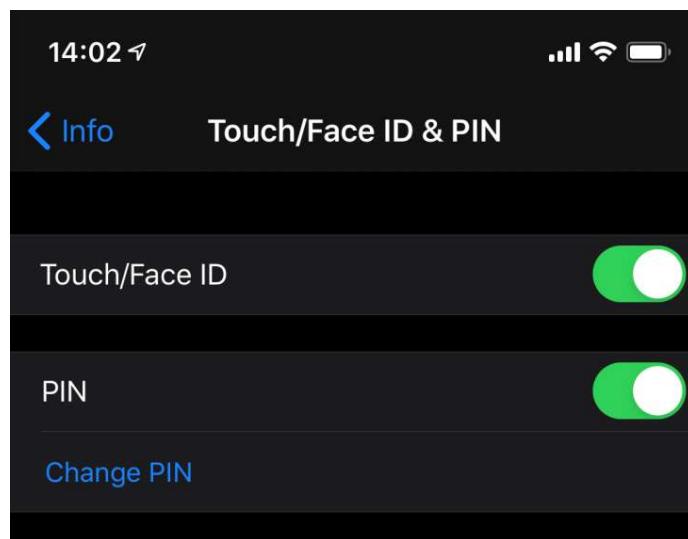
2. Tap on *Touch/Face ID & PIN*.



3. Enable and set up a 4-digit *PIN* for the application. The *PIN* is required to be enabled before you can enable *Touch/Face ID*.

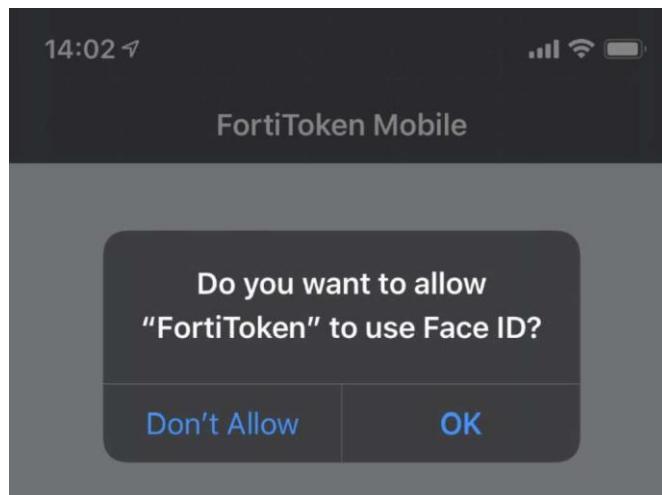


4. Enable *Touch/Face ID*.



You cannot enable *Touch/Face ID* for FortiToken if *Touch/Face ID* is not set up and enabled for device unlock (*iPhone Unlock* in this case) on iOS. You must first set up and enable *Touch/Face ID* from *Settings* on your iOS device.

5. When prompted by iOS, allow the FortiToken application to use *Touch/Face ID* by tapping on *OK* in the prompt.



## Applying multi-factor authentication

Multi-factor authentication (MFA) may also be set up for SSL VPN users, IPsec VPN users, administrators, firewall policy, wireless users, and so on.

# Authentication Methods

## Authentication Methods and Active Authentication

- Active
  - User receives a login prompt
  - Must manually enter credentials to authenticate
  - POP3, LDAP, RADIUS, Local, and TACACS+
- Passive
  - User does not receive a login prompt from FortiGate
  - Credentials are determined automatically
    - Method varies depending on type of authentication used
  - FSSO, RSSO, and NTLM

All the authentication methods you've learned about—local password authentication, server-based authentication, and two-factor authentication—use **active authentication**. Active authentication means that users are prompted to manually enter their login credentials before being granted access.

But not all users authenticate the same way. Some users can be granted access transparently, because user information is determined without asking the user to enter their login credentials. This is known as **passive authentication**. Passive authentication occurs with the single sign-on method for server-based password authentication: FSSO, RSSO, and NTLM.

# Firewall Policy - Source

## Firewall Policy—Source

- Firewall policies can use user and user group objects to define the source. The objects include:
  - Local firewall accounts
  - External (remote) server accounts
  - PKI (certificate) users
  - FSSO users
- Anyone who belongs to the group and provides correct information will have a successful authentication

**Policies & Objects > Firewall Policy**

Name	Full_Access
Incoming Interface	port3
Outgoing Interface	port1
Source	LOCAL_SUBNET External-Server-Users
Destination	all
Schedule	always
Service	ALL
Action	<input checked="" type="checkbox"/> ACCEPT <input type="checkbox"/> DENY

Select Entries

Address User Internet Service

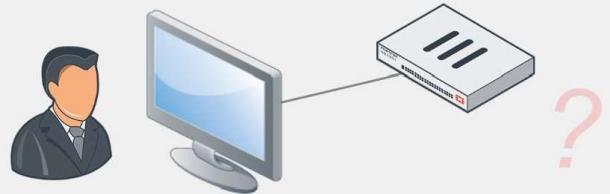
Q Search + Create

USER (2)

- Local (2)
- guest
- student

USER GROUP (3)

- External-Server-Users
- Guest-group
- SSO\_Guest\_Users



A firewall policy consists of access and inspection rules (compartmentalized sets of instructions) that tell FortiGate how to handle traffic on the interface whose traffic they filter. After the user makes an initial connection attempt, FortiGate checks the firewall policies to determine whether to accept or deny the communication session. However, a firewall policy also includes a number of other instructions, such as those dealing with authentication. You can use the source of a firewall policy for this purpose. **The source of a firewall policy must include the source address (IP address), but you can also include the user and user group.** In this way, any user, or user group that is included in the source definition for the firewall policy can successfully authenticate.

User and user group objects can consist of local firewall accounts, external server accounts, PKI users, and FSSO users.

# Protocols in Authentication

## Protocols

- A firewall policy must allow a protocol in order to show the authentication dialog that is used in active authentication:
  - HTTP
  - HTTPS
  - FTP
  - Telnet
- All other services are not allowed until the user has authenticated successfully through one of the protocols listed above

As well as the **DNS service**, the firewall policy must specify the allowed protocols, such as **HTTP**, **HTTPS**, **FTP**, and **Telnet**. If the firewall policy that has authentication enabled does not allow at least one of the supported protocols used for obtaining user credentials, the user will not be able to authenticate.

**Protocols are required for all authentication methods that use active authentication** (local password authentication, server-based password authentication, and two-factor authentication). Active authentication prompts the user for user credentials based on the following:

- **The protocol of the traffic**
- **The firewall policy**

**Passive authentication**, on the other hand, determines the user identity behind the scenes, and does not require any specific services to be allowed within the policy.

# Firewall Policy - Service

## Firewall Policy—Service

- DNS traffic can be allowed if user has not authenticated yet

- Hostname resolution is often required by the application layer protocol (HTTP/HTTPS/FTP/Telnet) that is used to authenticate
- DNS service must be explicitly listed as a service in the policy

### Policies & Objects > Firewall Policy

Name	Source	Destination	Schedule	Service	Action	NAT
port3 → port1 1						
Full_Access	External-Server-Users LOCAL_SUBNET	all	always	DNS HTTP	✓ ACCEPT	✓ Enabled

A firewall policy also checks the service in order to transport the named protocols or group of protocols. No service (with the exception of DNS) is allowed through the firewall policy before successful user authentication. DNS is usually used by HTTP so that people can use domain names for websites, instead of their IP address. DNS is allowed because it is a base protocol and will most likely be required to initially see proper authentication protocol traffic. Hostname resolution is almost always a requirement for any protocol. However, the DNS service must still be defined in the policy as allowed, in order for it to pass.

The screenshot shows the FortiGate Management UI with the 'Policy & Objects' section selected. Under 'Firewall Policy', a new policy is being created. The 'Service' field is highlighted with a red box. The 'Action' dropdown shows 'ACCEPT' is selected. The 'Additional Information' sidebar on the right provides links to online guides, documentation, and community forums.

In the example shown on this slide, policy ID 1 (Full\_Access) allows users to use external DNS servers in order to resolve host names, before successful authentication. DNS is also allowed if authentication is unsuccessful because users need to be able to try to authenticate again. Any service that includes DNS would function the same way, like the default ALL service.

HTTP service is TCP port 80 and does not include DNS (UDP port 53)

# Mixing Policies

## Mixing Policies

- Enabling authentication on a policy does not always force an active authentication prompt

port5 → port1									
ID	User Group	Source IP	Dest IP	Source Port	Dest Port	Action	Profile	Auth	Status
17	Guest	LOCAL_SUBNET	all			AV Guest_AV	SSL certificate-inspection	always	ALL ACCEPT Enabled
18	Contractor	LOCAL_SUBNET	all			AV Contractor_AV	SSL certificate-inspection	always	ALL ACCEPT Enabled
19	Other	LOCAL_SUBNET	all			AV default	SSL certificate-inspection	always	ALL ACCEPT Enabled

- Three options:
  - Enable authentication on every policy that could match the traffic
  - Enforce authentication on demand option (CLI option only)
  - Enable a captive portal on the ingress interface for the traffic
- If login cannot be determined passively, then FortiGate uses active authentication
  - FortiGate does not prompt the user for login credentials when it can identify the user passively
  - By default, active authentication is intended to be used as a backup when passive authentication fails

## Mixing Policies (All are just Active Authentications)

In the example shown on this slide, assuming active authentication is used:

- ✗ Any initial traffic from LOCAL\_SUBNET will not match **policy ID 17** (Guest). Policy ID 17 looks for both **IP** and **user**, and user group information (LOCAL\_SUBNET and Guest-group respectively), and since the user has not yet authenticated, the user group aspect of the traffic does not match. Since the policy match is not complete, FortiGate continues its search down the ID list, to see if there is a complete match.
- ✗ Next, FortiGate evaluates **policy ID 18** to see if the traffic matches. It will not for the same reason it did not match 17.
- ✓ Finally, FortiGate evaluates **policy ID 19** to see if the traffic matches. It matches all criteria, so traffic is allowed with no need to authenticate.

When you use only active authentication, if all possible policies that could match the source IP have authentication enabled, then the user will receive a login prompt (assuming they use an acceptable login protocol). In other words, if policy ID 19 also had authentication enabled, the users would receive login prompts.



## Solutions to force FortiGate to send Login Prompt to the users:

### Three options:

- Enable authentication on every policy that could match the traffic
- Enforce authentication on demand option (CLI option only)
- Enable a captive portal on the ingress interface for the traffic

## Mixing Policies (All are just Passive Authentications)

If you use passive authentication and it can successfully obtain user details, then traffic from LOCAL\_SUBNET with users that belong to Guest-group will apply to policy ID 17, even though policy ID 19 does not have authentication enabled.

## Mixing Policies (Both Active & Passive Authentications)

If you use both active and passive authentication, and FortiGate can identify a user's credentials through passive authentication, the user never receives a login prompt, regardless of the order of any firewall policies. This is because there is no need for FortiGate to prompt the user for login credentials when it can identify who the user is passively. **When you combine active and passive authentication methods, active authentication is intended to be used as a backup, to be used only when passive authentication fails.**

# Active Authentication Behavior

## Active Authentication Behavior

- Enable authentication on every policy that could match the traffic:
  - All firewall policies must have authentication enabled (active or passive)
- If there is a fall-through policy in place, unauthenticated users are not prompted for authentication
- Enforce authentication on-demand option:
  - CLI option only

```
# config user setting  
(setting) # set auth-on-demand <always|implicitly>
```

- Provides more granular control
  - Authentication is enabled at a firewall policy level
- You must place passive authentication policies on top of active authentication policies

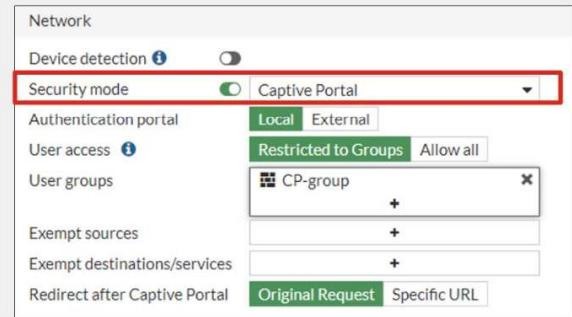
As mentioned earlier, there are three different ways you can alter active authentication behavior.

If you have an active authentication firewall policy followed by a fall-through policy that does not have authentication enabled on it, then all traffic will use the fall-through policy. This means that users are not asked to authenticate. By default, all traffic passes through the catch-all policy without being authenticated. You can alter this behavior:

- By enabling authentication **on all firewall policies**. When you enable authentication, all the systems must authenticate before traffic is placed on the egress interface.
- Alternatively, only on the CLI, you can change the **auth-on-demand** options. There are two options:
  - **Implicitly** The default option. It will not trigger authentication if there is a fall through policy.
  - **Always** Triggers an authentication prompt for policies that have active authentication enabled regardless of a fall-through policy. In this case, the traffic is not allowed until authentication is successful.
- Enable a **captive portal** on the ingress interface for the traffic.

If you want to have all users connect to a specific interface, then it is better to enable captive portal authentication at the interface level. This way, all devices must authenticate before they are allowed to access any resources

- Enable a captive portal on the ingress interface for the traffic:
  - Authentication happens at an interface level
  - Traffic is not allowed without valid authentication unless it matches an exemption
  - All users are prompted for authentication before they can access any resource



The screenshot shows the 'Edit Interface' configuration page. On the left, the 'Interfaces' tab is selected. In the main area, under 'IPv4' settings, various protocols are checked: HTTPS, FMG-Access, SSH, PING, and SNMP. Under 'Network' settings, 'Device detection' is off, and 'Security mode' is set to 'Captive Portal'. The 'Captive Portal' dropdown has 'Local' selected. Below these, there are sections for 'User access', 'Exempt sources', 'Exempt destinations/services', and 'Redirect after Captive Portal'.

# Captive Portal

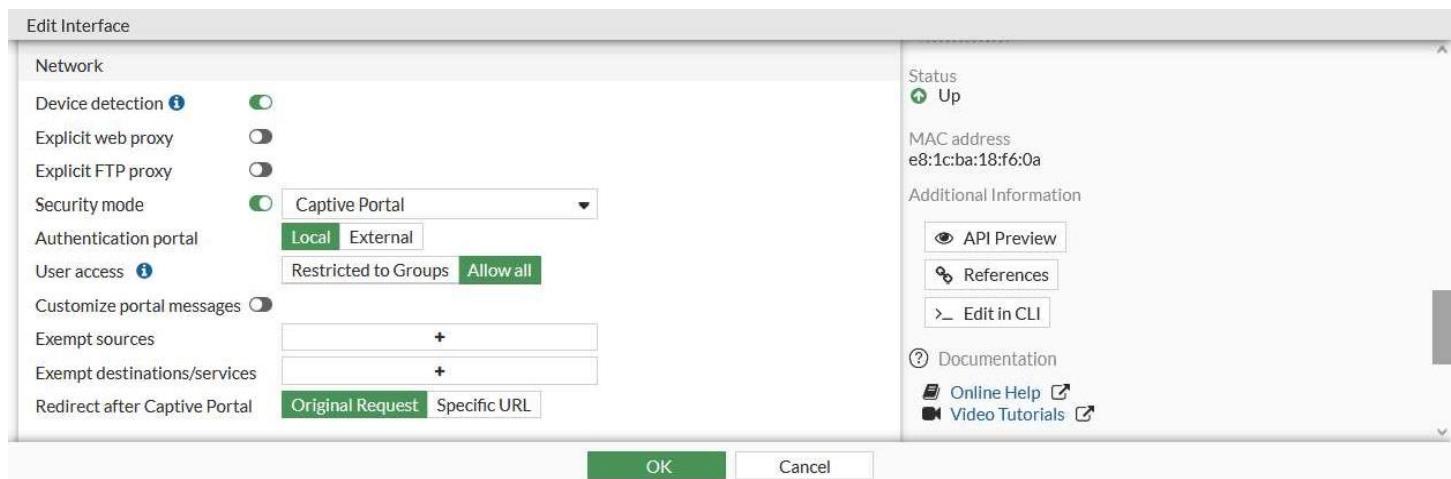
A captive portal is used to enforce authentication before web resources can be accessed. Until a user authenticates successfully, any HTTP request returns the authentication page. After successfully authenticating, a user can access the requested URL and other web resources, as permitted by policies. The captive portal can also be configured to only allow access to members of specific user groups.

Captive portals can be hosted on the FortiGate or an external authentication server. They can be configured on **any network interface**, including VLAN and WiFi interfaces. On a WiFi interface, the access point appears open, and the client can connect to access point with no security credentials, but then sees the captive portal authentication page.

All users on the interface are required to authenticate. Exemption lists can be created for devices that are unable to authenticate, such as a printer that requires access to the internet for firmware upgrades.

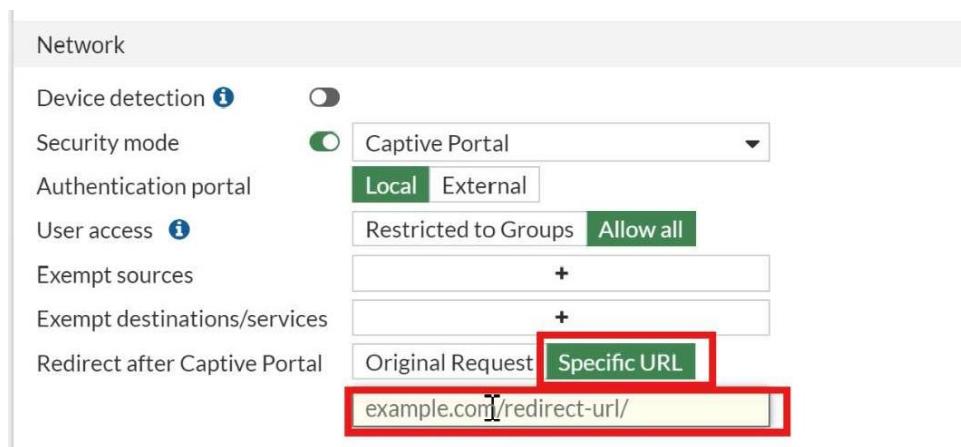
## To configure a captive portal in the GUI:

1. Go to **Network > Interfaces** and edit the interface that the users connect to. The interface **Role** must be **LAN** or **Undefined**.
2. Enable **Security mode**.



3. Configure the following settings, then click **OK**.

<b>Authentication Portal</b>	Configure the location of the portal:
	<ul style="list-style-type: none"> <li><i>Local</i>: the portal is hosted on the FortiGate unit.</li> <li><i>External</i>: enter the FQDN or IP address of external portal.</li> </ul>
<b>User access</b>	Select if the portal applies to all users, or selected user groups:
	<ul style="list-style-type: none"> <li><i>Restricted to Groups</i>: restrict access to the selected user groups. The <i>Login page</i> is shown when a user tries to log in to the captive portal.</li> <li><i>Allow all</i>: all users can log in, but access will be defined by relevant policies. The <i>Disclaimer page</i> is shown when a user tried to log in to the captive portal.</li> </ul>
<b>Customize portal messages</b>	Enable to use custom portal pages, then select a replacement message group.
<b>Exempt sources</b>	<p>Select sources that are exempt from the captive portal.</p> <p>Each exemption is added as a rule in an automatically generated exemption list.</p>
<b>Exempt destinations/services</b>	<p>Select destinations and services that are exempt from the captive portal.</p> <p>Each exemption is added as a rule in an automatically generated exemption list.</p>
<b>Redirect after Captive Portal</b>	<p>Configure website redirection after successful captive portal authentication:</p> <ul style="list-style-type: none"> <li><i>Original Request</i>: redirect to the initially browsed to URL.</li> <li><i>Specific URL</i>: redirect to the specified URL.</li> </ul>



## Custom captive portal pages

Portal pages are HTML files that can be customized to meet user requirements.

Most of the text and some of the HTML in the message can be changed. Tags are enclosed by double percent signs (%%); most of them should not be changed because they might carry information that the FortiGate unit needs.

The images on the pages can be replaced. For example, your organization's logo can replace the Fortinet logo.

The following pages are used by captive portals:

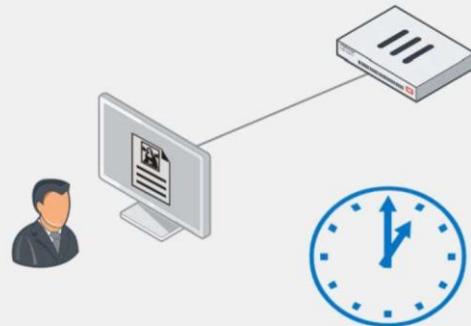
<b>Login Page</b>	Requests user credentials.  The %%QUESTION%% tag provides the <i>Please enter the required information to continue.</i> text.  This page is shown to users that are trying to log in when <i>User access</i> is set to <i>Restricted to Groups</i> .
<b>Login Failed Page</b>	Reports that incorrect credentials were entered, and requests correct credentials.  The %%FAILED_MESSAGE%% tag provides the <i>Firewall authentication failed. Please try again.</i> text.
<b>Disclaimer Page</b>	A statement of the legal responsibilities of the user and the host organization that the user must agree to before proceeding. This page is shown users that are trying to log in when <i>User access</i> is set to <i>Allow all</i> .
<b>Declined Disclaimer Page</b>	Shown if the user does not agree to the statement on the Disclaimer page. Access is denied until the user agrees to the disclaimer.

# Authentication Timeout

## Authentication Timeout

```
#config user setting  
    set auth-timeout-type [idle-timeout|hard-timeout|new-session]  
end
```

- Timeout specifies how long a user can remain idle before the user must authenticate again
  - Default is 5 minutes
- Three options for behavior:
  - Idle (default): no traffic for that amount of time
  - Hard: authentication expires after that amount of time, regardless of activity
  - New session: authentication expires if no new session is created in that amount of time



An authentication timeout is useful for security purposes. It minimizes the risk of someone using the IP of the legitimate authenticated user. It also ensures users do not authenticate and then stay in memory indefinitely. If users stayed in memory forever, it would eventually lead to memory exhaustion.

### There are three options for timeout behavior:

- **Idle:** The idle timer starts when a user initiates a session. As long as data are transferred in this session, the timer continually resets. If the data flow stops, the timer is allowed to advance until it reaches its limit. When the user has been idle for too long, they must re-authenticate before traffic is allowed to continue in that session. This is the default setting. It can be configured in the [GUI and CLI](#).
- **Hard:** The hard timer starts when a user initiates a session. Regardless of the user's behavior, when the timeout is reached, all the sessions for that user must be re-authenticated. This timeout is not affected by any events. This setting can be configured [in the CLI](#).
- **New session:** The session timer starts when a user initiates a session. When the timeout is reached, existing sessions may continue. New sessions are not allowed until the user re-authenticates. This timeout is not affected by any events.  
This setting can be configured [in the CLI](#).

Choose the type of timeout that best suits the authentication needs of your environment.

## To configure timeout for authenticated users:

```
config user setting
    set auth-timeout-type {idle-timeout | hard-timeout | new-session}
        set auth-timeout <integer>
end
```

## To configure the authentication timeout for a user group:

```
config user group
    edit <name>
        set authtimeout <integer>
    next
end
```

- Enter the desired timeout, in minutes, from 1 to 1440 (24 hours). The default time is 5 minutes. Only idle timeout can be configured in the GUI.

# Monitoring Users

The screenshot shows the 'Firewall Users' page from a dashboard. It displays two summary cards: one for 'Method' (Firewall) and one for 'User Group' (CP-group), both showing 1 user. Below these are search and filter fields, and a table with a single row for a user named 'student'. A red arrow points from the 'Deauthenticate' button in the table to a confirmation dialog box titled 'Confirm' that asks if the user wants to deauthenticate the selected user(s). The dialog has 'OK' and 'Cancel' buttons.

User Name	IP Address	User Group	Duration	Traffic Volume	Method
student	10.0.1.10	CP-group	1 minute(s) and 9 second(s)	10.43 kB	Firewall

You can monitor users who authenticate through your firewall policies using:

**Dashboard > Assets & Identities > Firewall Users page.** It displays the user, user group, duration, IP address, traffic volume, and authentication method.

It does not include **administrators**, because they are not authenticating through firewall policies that allow traffic. They are logging in directly on FortiGate.

This page also allows you to **disconnect a user**, or multiple users, at the same time.

# LAB

In this lab, you will examine how to configure FortiGate to communicate with remote LDAP and RADIUS servers for server-based password authentication.

## Objectives

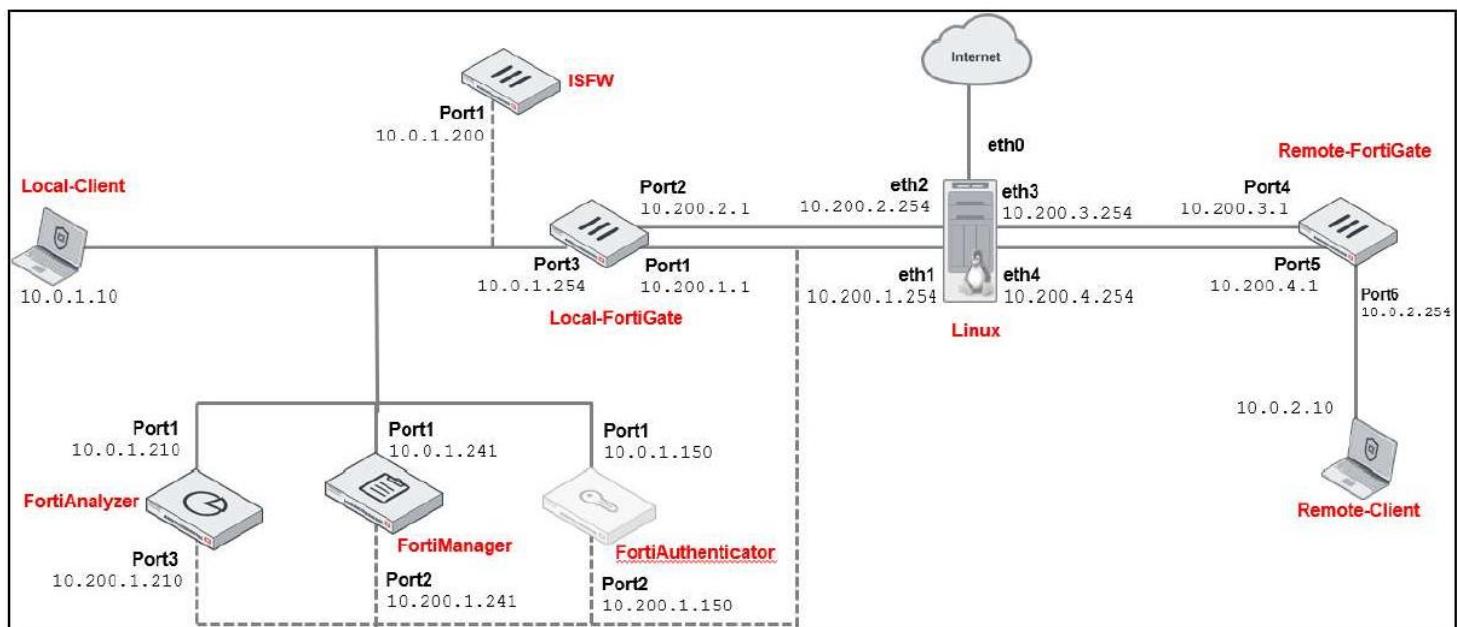
- Configure server-based password authentication with an LDAP server
- Configure server-based password authentication with a RADIUS server

We have two exercises in this LAB:

**Exercise 1: Configuring an LDAP Server on FortiGate**

**Exercise 2: Configure a RADIUS Server on FortiGate**

## LAB Topology:



# Exercise 1: Configuring an LDAP Server

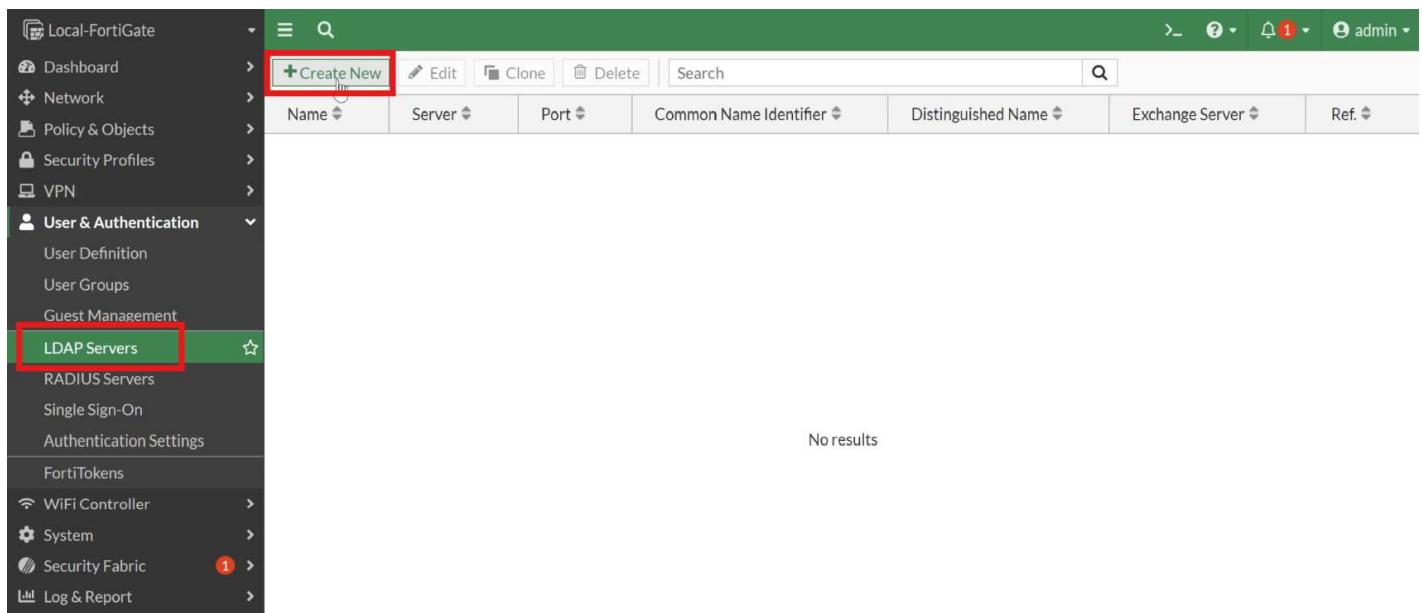
In this exercise, you will examine how to configure an LDAP server on FortiGate for remote authentication, create a remote authentication group for remote users, and then add that group as a source in a firewall policy. Finally, you will authenticate as one of the remote users, and then monitor the login as the administrator.

## Configure an LDAP Server on FortiGate

You will configure FortiGate to point to a preconfigured FortiAuthenticator acting as an LDAP server for server-based password authentication.

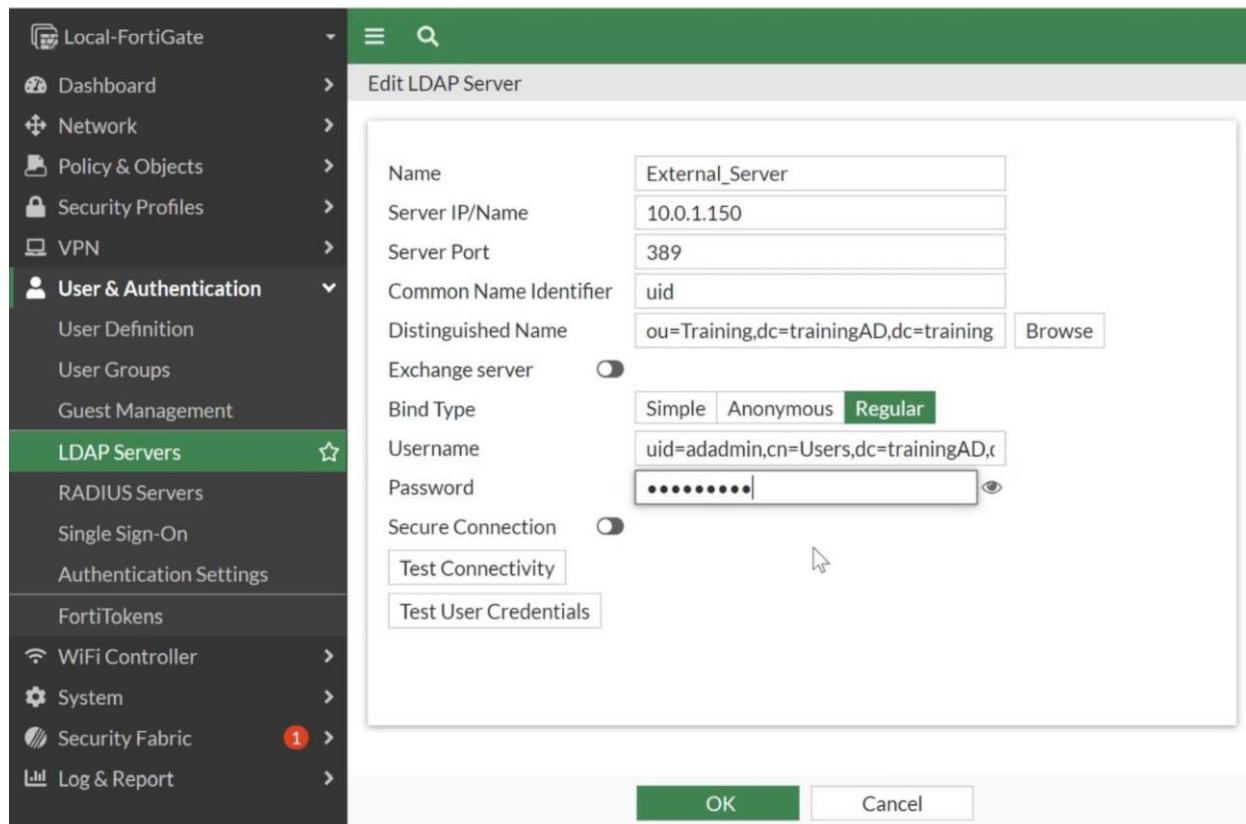
### To configure an LDAP server on FortiGate

1. Connect to the Local-FortiGate GUI, and then log in with the username **admin** and password **password**.
2. Click **User & Authentication > LDAP Servers**, and then click **Create New**.
3. Configure a server using the following settings:

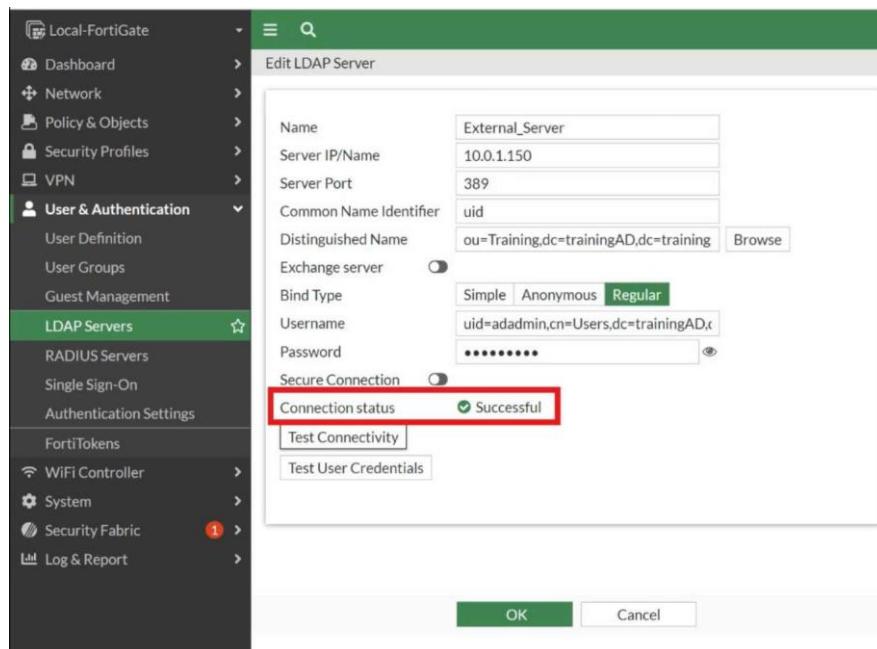


The screenshot shows the FortiGate Local-FortiGate GUI interface. The left sidebar menu is visible, showing various sections like Dashboard, Network, Policy & Objects, Security Profiles, VPN, User & Authentication, LDAP Servers (which is highlighted with a red box), RADIUS Servers, Single Sign-On, Authentication Settings, FortiTokens, WiFi Controller, System, Security Fabric (with a red circle containing the number 1), and Log & Report. The main content area has a green header bar with the title 'Local-FortiGate' and user information ('admin'). Below the header is a search bar with a 'Create New' button (also highlighted with a red box). The main table has columns for Name, Server, Port, Common Name Identifier, Distinguished Name, Exchange Server, and Ref. A message 'No results' is displayed at the bottom of the table area.

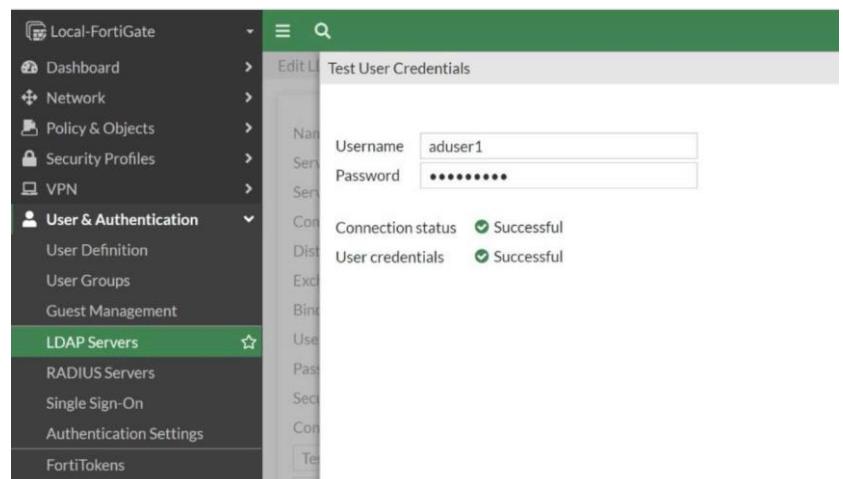
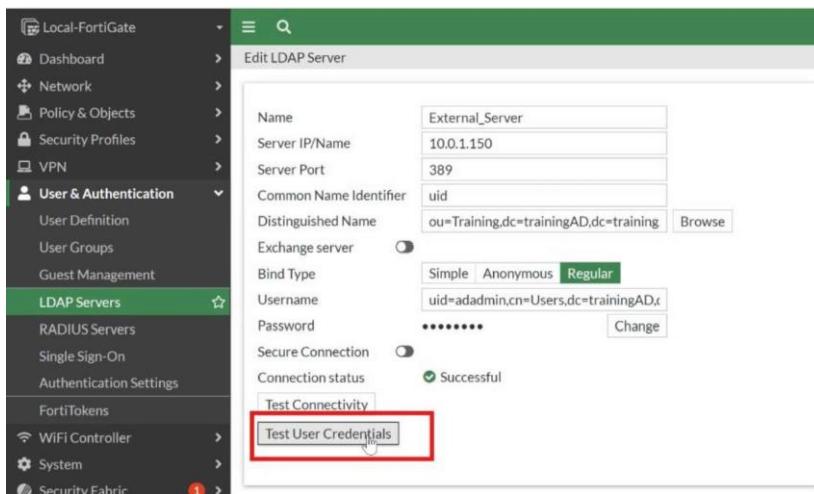
Field	Value
Name	External_Server
Server IP/Name	10.0.1.150 This is the IP address of the <b>FortiAuthenticator</b> acting as the <b>LDAP server</b> .
Server Port	389 This is the default port for LDAP.
Common Name Identifier	uid This is the attribute name used to find the username on the preconfigured LDAP server.
Distinguished Name	ou=Training,dc=trainingAD,dc=training,dc=lab This is the domain name for the LDAP directory on FortiAuthenticator, with all users located under the <b>Training</b> organizational unit (ou).
Bind Type	Regular
Username	uid=adadmin,cn=Users,dc=trainingAD,dc=training,dc=lab You are using the credentials of an LDAP user called <b>adadmin</b> to authenticate to the LDAP server.
Password	Training! This is the password preconfigured for the <b>adadmin</b> user. You must use it to be able to bind.



#### 4. Click Test Connectivity.



You should see a message indicating that the connection was successful.



#### 5. Click OK.

Name	Server	Port	Common Name Identifier	Distinguished Name	Exchange Server	Ref.
External_Server	10.0.1.150	389	uid	ou=Training,dc=trainingAD,dc=training,dc=lab		0

## Assign an LDAP User Group to a Firewall Group

You will assign an LDAP user group (**AD\_users**) that includes two users (**aduser1** and **aduser2**) to a firewall user group, called **Remote-users**, on FortiGate. By doing this, you will be able to configure firewall policies to act on the firewall user group.

Usually, groups are used to more effectively manage individuals who have a shared relationship.



The **Remote-users** firewall group is preconfigured for you. However, you must modify it to add the users from the remote LDAP server you configured in the previous procedure.

### Take the Expert Challenge!

On Local-FortiGate (10.0.1.254), assign the Active Directory user group called **AD\_users** to the FortiGate firewall user group called **Remote-users**.

If you require assistance, or to verify your work, use the step-by-step instructions that follow.

## To assign a user to a user group

1. On the Local-FortiGate GUI, click **User & Authentication > User Groups**, and then edit the **Remote-users** group.

Notice that it's currently configured as a firewall group.

2. In the **Remote Groups** table, click **Add** to add users from the remote LDAP server.

The screenshot shows the FortiGate management interface. On the left, the navigation menu is open, showing various sections like Dashboard, Network, Policy & Objects, Security Profiles, VPN, User & Authentication, and User Groups. The User Groups section is currently selected.

The main pane displays a table of user groups:

Group Name	Group Type	Members	Ref.
Guest-group	Firewall	guest	0
Remote-users	Firewall	guest	0
SSO_Guest_Users	Fortinet Single Sign-On (FSSO)		1
Training	Firewall		0

A modal dialog titled "Edit User Group" is open over the table. It contains fields for "Name" (Remote-users), "Type" (Firewall), and "Members". Below these fields is a "Remote Groups" section with a table:

Remote Server	Group Name
No results	

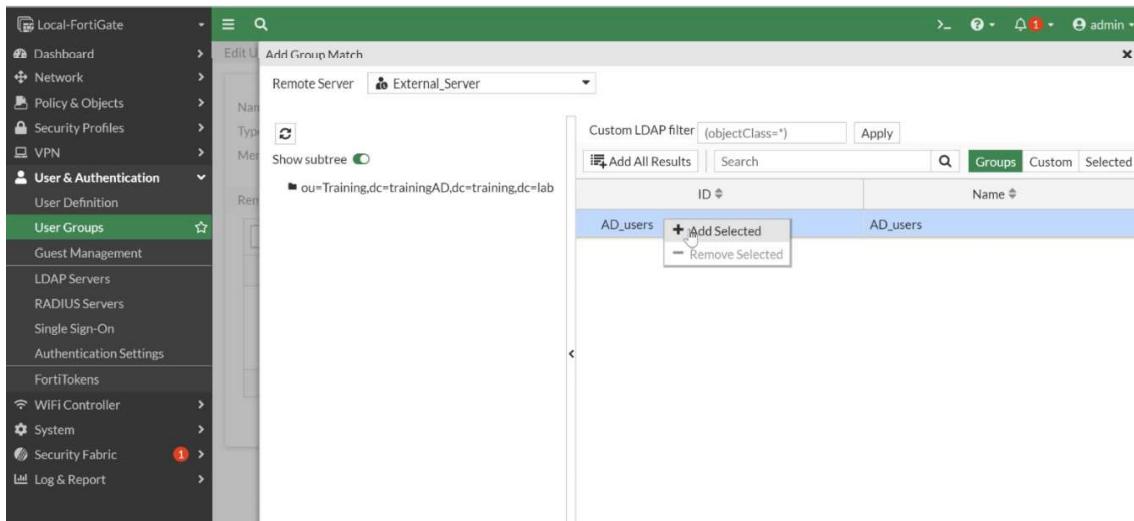
At the bottom of the dialog are "OK" and "Cancel" buttons.

The **Add Group Match** window opens.

### 3. In the **Remote Server** field, select **External\_Server**.

The screenshot shows the "Add Group Match" dialog box. The "Remote Server" dropdown menu is open, displaying two options: "LDAP SERVER (1)" and "External\_Server". The "External\_Server" option is highlighted with a gray background, indicating it is selected.

- On the **Groups** tab, right-click **AD\_users**, and then click **Add Selected**.



**AD\_users** has a green check mark beside it, which indicates that it was added.

The screenshot shows the 'Selected' tab in the FortiGate interface. The 'AD\_users' group is listed with a green checkmark icon next to it, indicating it is selected.

- Click **OK**.

The users in this Active Directory group are now included in the FortiGate **Remote-users** firewall user group. Only users from the remote LDAP server that match this user group entry can authenticate.

The screenshot shows the 'Edit User Group' dialog for the 'Remote-users' group. The 'Members' field contains '+ AD\_users'. The 'Remote Groups' section shows a table with one entry: 'External\_Server' with 'cn=AD\_users,ou=Training,dc=trainingAD,dc=training,dc=...'. This entry is highlighted with a red border.

- Click **OK**.

# Add the Remote User Group to the Firewall Policy

Now that you have added the LDAP server to the **Remote-users** firewall user group, you can add the group to a firewall policy. This allows you to control access to network resources, because policy decisions are made for the group as a whole.

## To add the remote user group to the firewall policy

1. On the Local-FortiGate GUI, click **Policy & Objects > Firewall Policy**, and then double-click the existing port3 to port1 firewall policy.

The screenshot shows the 'Firewall Policy' section of the FortiGate GUI. The left sidebar has 'Policy & Objects' expanded, with 'Firewall Policy' selected. The main area shows a table of policies. One policy, 'Full\_Access', is highlighted. Its details are shown in a modal: Name: Full\_Access, Incoming Interface: port3, Outgoing Interface: port1, Source: LOCAL\_SUBNET, Destination: all, Schedule: always, Service: ALL, Action: ACCEPT, NAT: checked, Type: Standard, Security Profile: SSL certificate. Below the table is a note: '+ Implicit 1'.

2. Configure the following setting:

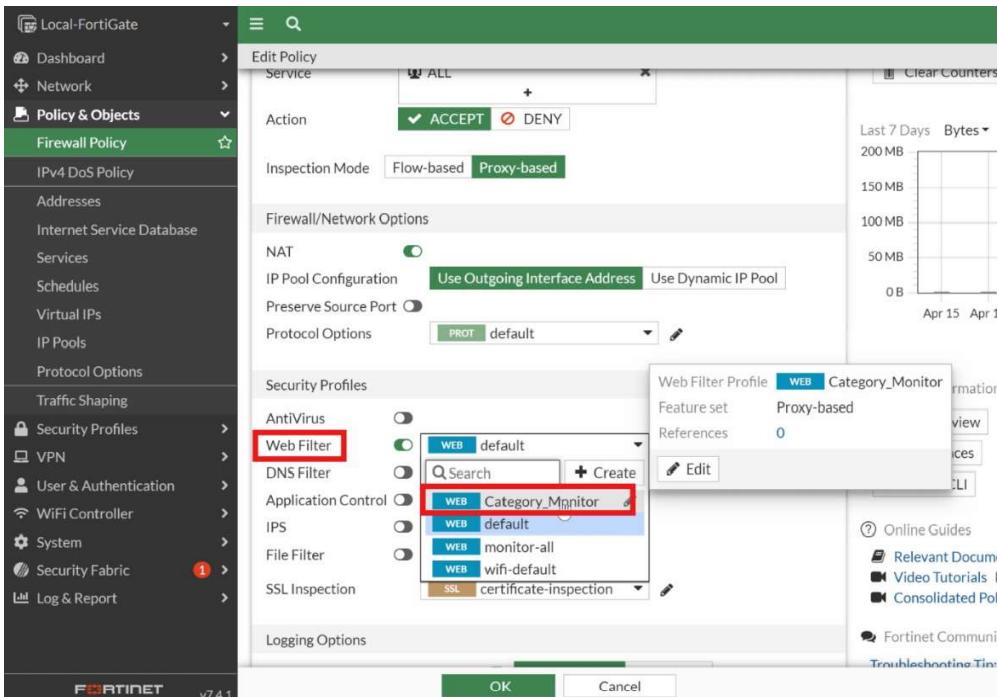
Field	Value
Source	Click <b>+</b> , and then select <b>Remote-users</b> (located under <b>User</b> ).

The screenshot shows the 'Edit Policy' dialog for the 'Full\_Access' policy. The 'Source' field is set to 'LOCAL SUBNET' and contains 'Remote-users'. A 'Select Entries' dialog is open, showing the 'User' tab with 'Remote-users' selected. The 'OK' button is at the bottom of the dialog.

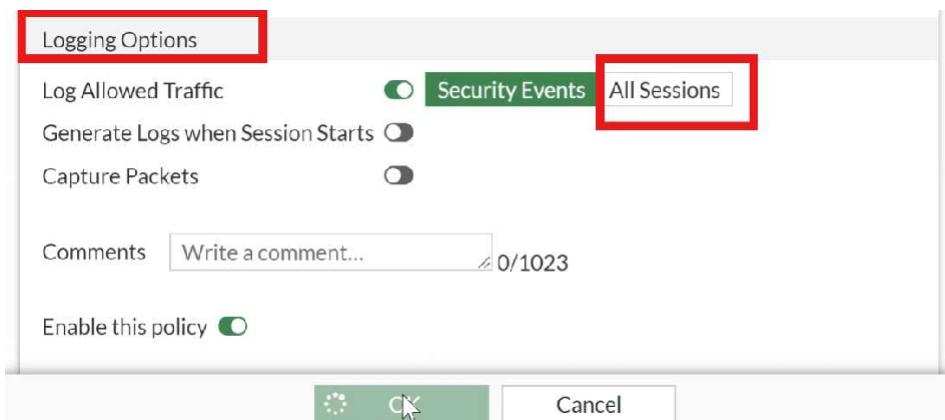
3. In the **Security Profiles** section, enable **Web Filter**, and then select **Category\_Monitor**.

This web filter was preconfigured and is set to block the following categories:

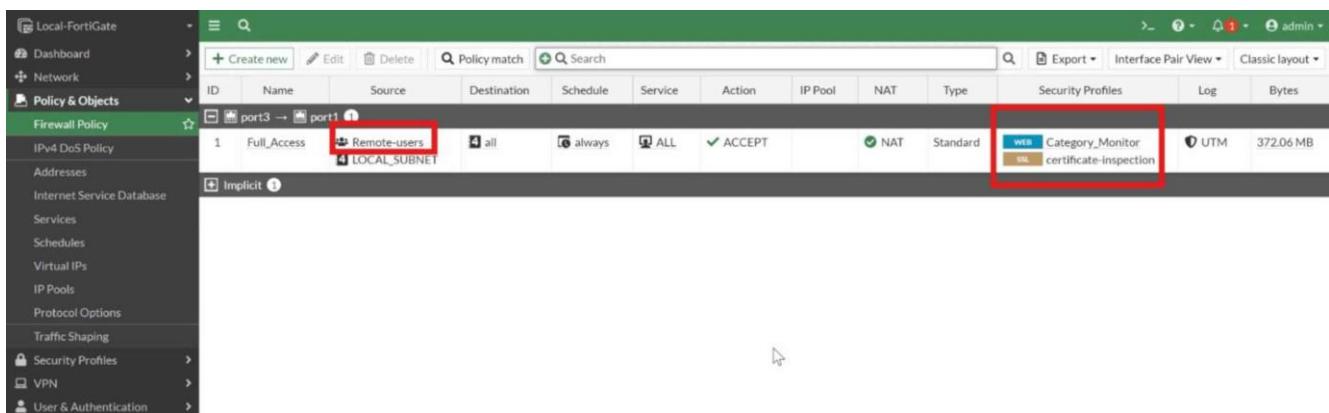
### Potentially Liable, Adult/Mature Content, and Security Risk.



4. In the **Logging Options** section, ensure **Log Allowed Traffic** is enabled, and then select **All Sessions**.



5. Click **OK**.



## To test whether aduser1 can successfully authenticate

1. On the Local-FortiGate CLI, log in with the username **admin** and password **password**.
2. Enter the following command:

```
diagnose test authserver ldap <LDAP server name> <LDAP user name> <password>
```

Where:

- <LDAP server name> is **External\_Server** (case sensitive)
- <LDAP user name> is **aduser1**
- <password> is **Training!**

A message like the following example should appear to indicate that authentication was successful:

```
Trying 127.0.0.1...
Connected to 127.0.0.1.
Escape character is '^']'.

Local-FortiGate login: admin
Password:
Welcome!

Local-FortiGate #
Local-FortiGate #
Local-FortiGate #
Local-FortiGate # diagnose test authserver ldap External_Server aduser1 Training!
authenticate 'aduser1' against 'External_Server' succeeded!
Group membership(s) - cn=AD_users,ou=Training,dc=trainingAD,dc=lab
Local-FortiGate #
Local-FortiGate #
Local-FortiGate # █
```

3. Close the Local-FortiGate CLI window.

## Authenticate and Monitor the Authentication

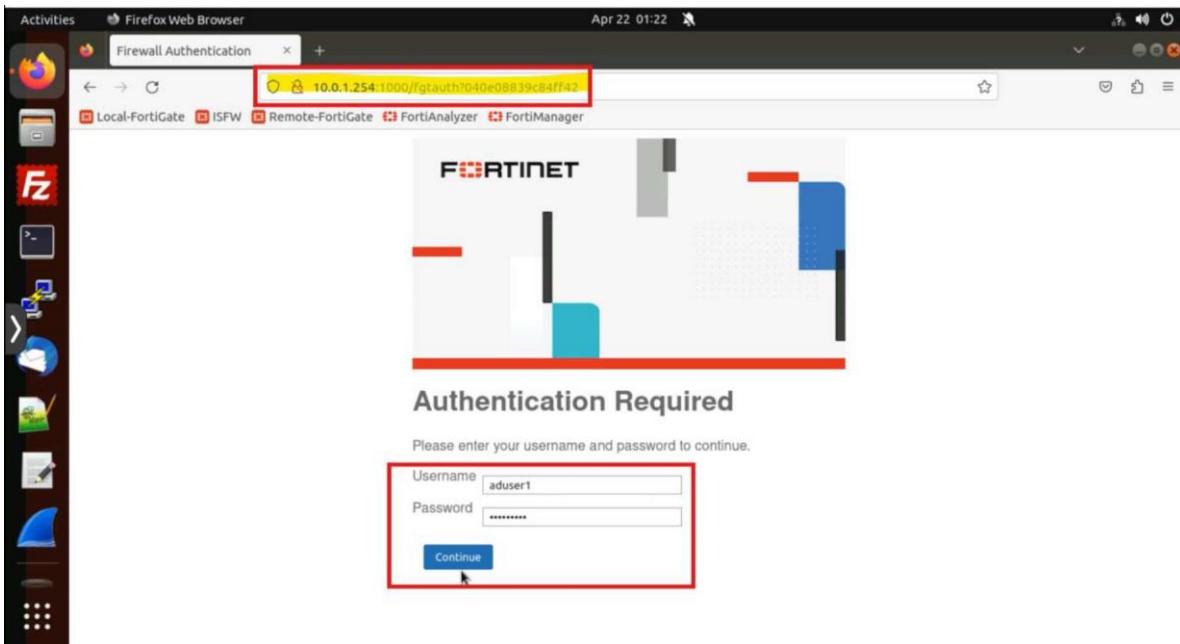
You will authenticate through the firewall policy as **aduser1**. This user is a member of the **Remote-users** group on FortiGate. Then, you will monitor the authentication.

## To authenticate as a remote user

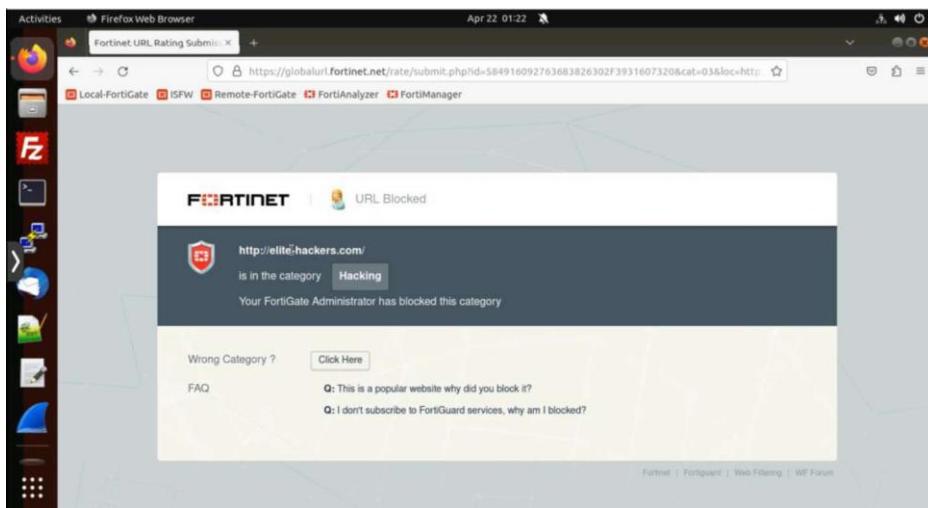
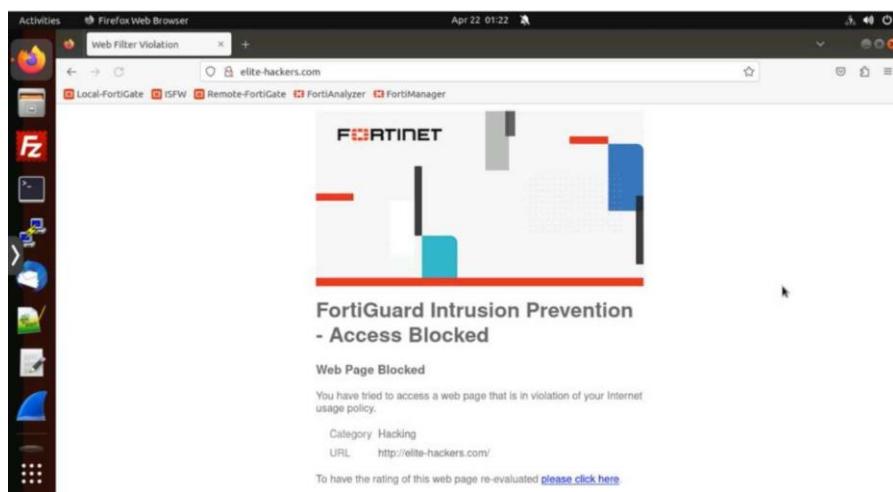
1. On the Local-Client VM, open a new browser tab, and then go to [elite-hackers.com](http://elite-hackers.com).

You are asked to log in to the network.

2. Log in as **aduser1** with the password **Training!**.



This URL is set to be blocked by the web filter security profile you enabled in the firewall policy.



Notice that the blocked page displays a **replacement message** that includes useful information, such as the **URL** and **Category**.

# To monitor active authenticated users

1. Return to the browser tab where you are logged in to the Local-FortiGate GUI as **admin**.
2. Click **Dashboard > Assets&Identities**, and then click **Firewall Users** to expand it to full screen to view this login authentication and monitor the firewall authenticated user.

The screenshot shows the Local-FortiGate GUI interface. On the left, the navigation menu is open, showing the 'Assets & Identities' section selected. Under 'Assets & Identities', 'WiFi' is expanded, and 'Firewall Users' is listed under it. A red box highlights the 'Firewall Users' link. The main dashboard area has a title 'Firewall Users' with a count of '1'. Below this is a large circular gauge chart with a green center and a grey outer ring, also showing a value of '1'. A tooltip says 'Click to expand'. At the bottom of the dashboard, there are sections for 'Quarantine' and 'Category'.

You will see **aduser1** listed along with other information, such as **User Group** and **IP Address**.

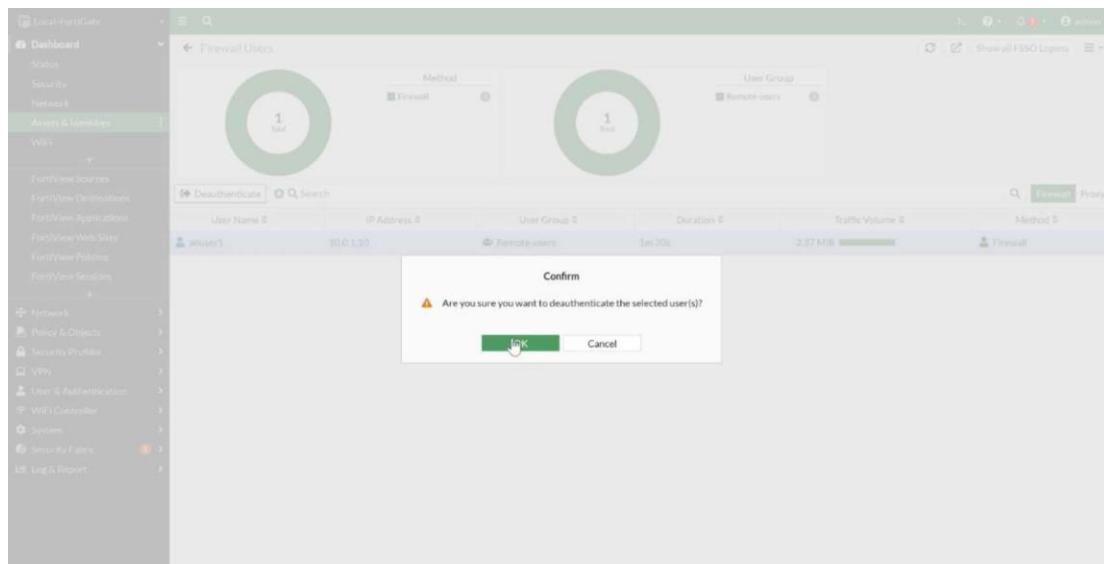
3. Click **aduser1**, and then click **Deauthenticate**.

This screenshot shows the 'Firewall Users' dashboard with one user listed: 'aduser1'. The user details shown are IP Address (10.0.1.10), User Group (Remote-users), Duration (1m 30s), and Traffic Volume (2.37 MiB). A red box highlights the 'Deauthenticate' button, which is located in the top row of the user list. The rest of the interface includes search and filter options like 'User Name', 'IP Address', 'User Group', 'Duration', 'Traffic Volume', and 'Method'.



The **config user setting** CLI command determines how long a user can remain authenticated. However, you can choose to manually revoke a user authentication by selecting the user in the **Firewall User Monitor** list, and then clicking **Deauthenticate**. After the user is deauthenticated, the user disappears from the list, because it is reserved for active users only.

4. In the **Confirm** window, click **OK**.



This deauthenticates the user. The user must log in again to access the resources that the firewall policy protects.

## Remove the User Group from the Firewall Policy

You will remove the user group assigned to the firewall policy for authentication.

### To remove the remote user group from the firewall policy

1. On the Local-FortiGate GUI, click **Policy & Objects > Firewall Policy**, and then double-click the existing port3 to port1 firewall policy.

ID	Name	From	To	Source	Destination	Schedule	Service	Action
1	Full_Access	port3	port1	Remote-users LOCAL_SUBNET	all	always	ALL	✓ ACCEPT

2. In the **Source** field, remove the **Remote-users** user group.

Edit Policy

Name	Full_Access
Incoming Interface	port3
Outgoing Interface	port1
Source	<input checked="" type="checkbox"/> LOCAL_SUBNET <input checked="" type="checkbox"/> Remote-users
Destination	<input checked="" type="checkbox"/> all
Schedule	always
Service	ALL
Action	<input checked="" type="checkbox"/> ACCEPT <input type="checkbox"/> DENY

3. Click **Close**, and then click **OK** to save the changes.

# Exercise 2: Configuring a RADIUS Server

In this exercise, you will examine how to configure a RADIUS server on FortiGate for remote authentication, create a remote authentication group for remote users, and then add that group as a source in a firewall policy. Finally, you will authenticate as one of the remote users, and then monitor the login as the administrator.

## Configure a RADIUS Server on FortiGate

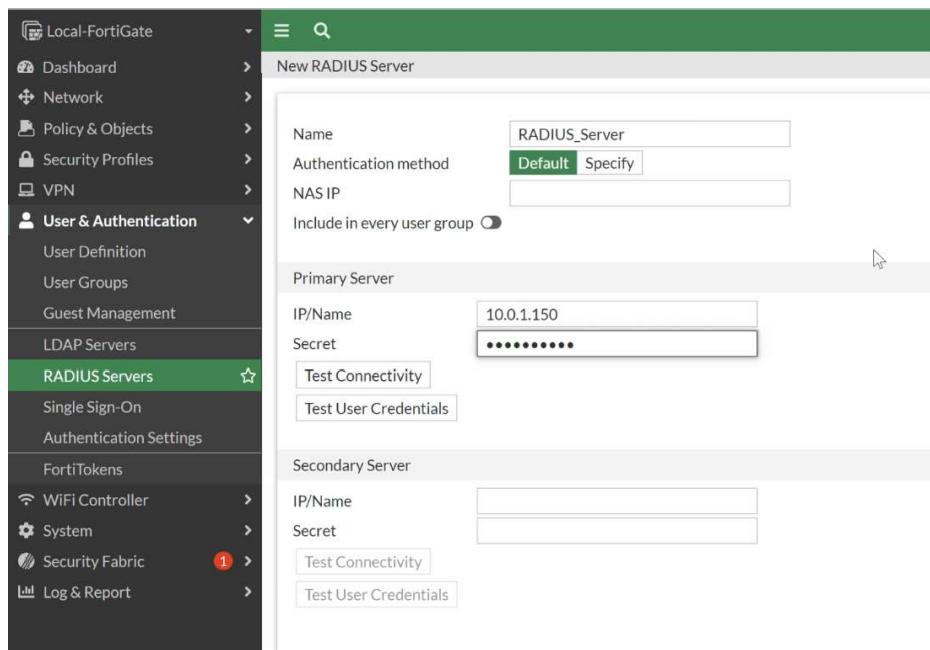
You can configure FortiGate to point to a preconfigured FortiAuthenticator acting as a RADIUS server for server-based password authentication.

### To configure a RADIUS server on FortiGate

1. Connect to the Local-FortiGate GUI, and then log in with the username **admin** and password **password**.
2. Click **User & Authentication > RADIUS Servers**, and then click **Create New**.
3. Configure a server using the following settings:

Field	Value
Name	RADIUS_Server
Authentication method	Default
Primary Server IP/Name	10.0.1.150 <b>This is the IP address of the FortiAuthenticator acting as the RADIUS server.</b>
Secret	Training1!

4. Click **Test Connectivity**.



You should see a message indicating that the connection was successful.

The screenshots show the configuration and testing of a RADIUS server on a FortiGate device. The top part shows the initial setup with a primary server at 10.0.1.150. The bottom part shows the successful test of user credentials 'radius1' with password '\*\*\*\*\*', resulting in a successful connection and user credentials.

5. Click **OK**.

# Assign a RADIUS User Group to a Firewall Group

You will assign a RADIUS user group (**Training**) that includes a user (**radius1**) to a firewall user group, called **Training**, on FortiGate. By doing this, you will be able to configure firewall policies to act on the firewall user group.

Usually, groups are used to more effectively manage individuals who have a shared relationship.



The **Training** firewall group is preconfigured for you. However, you must modify it to add the users from the remote RADIUS server you configured in the previous procedure.

## Take the Expert Challenge!

On Local-FortiGate ([10.0.1.254](#)), assign the RADIUS user group called **Training** to the FortiGate firewall user group called **Training**.

If you require assistance, or to verify your work, use the step-by-step instructions that follow.

## To assign a user to a user group

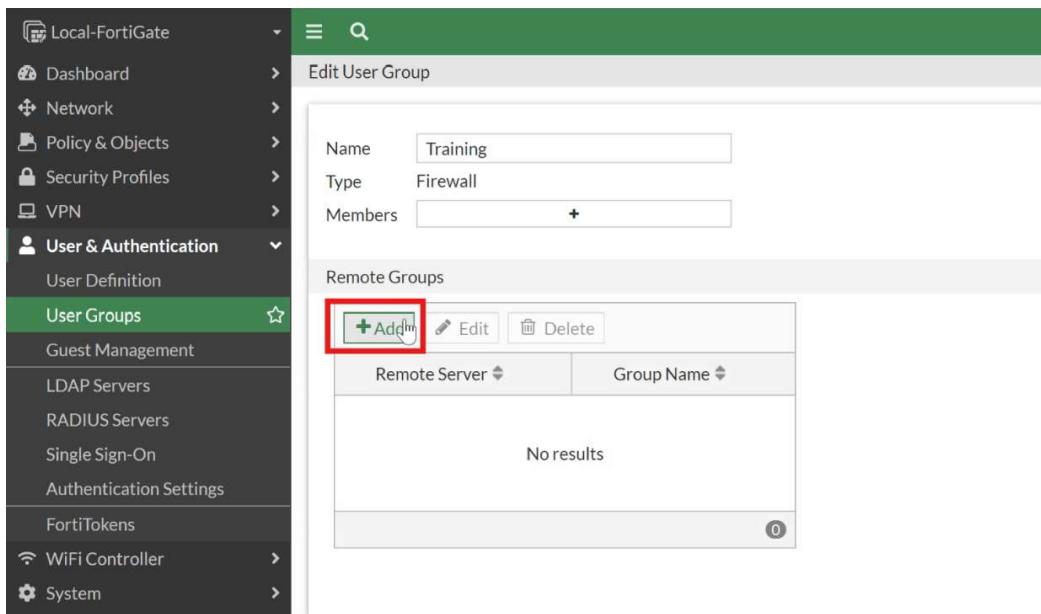
1. On the Local-FortiGate GUI, click **User & Authentication > User Groups**, and then edit the **Training** group.

Notice that it's currently configured as a firewall group.

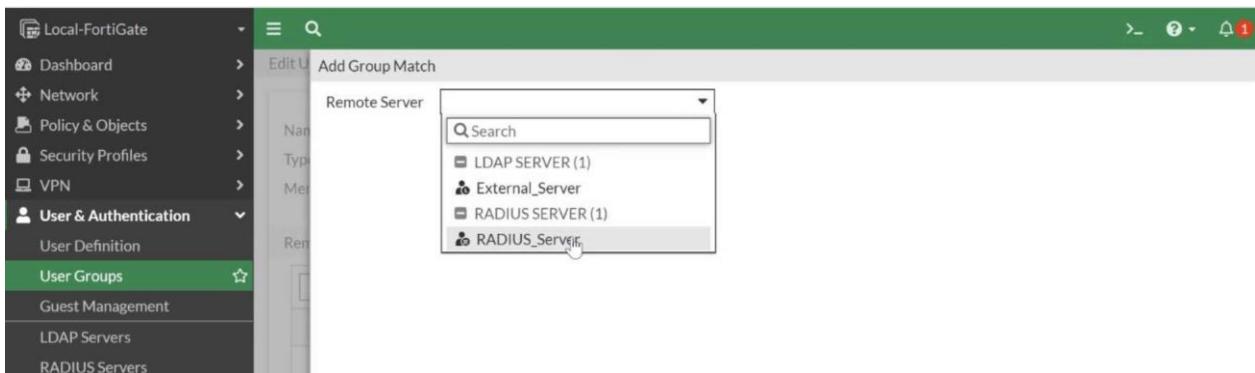
2. In the **Training** table, click **Add** to add users from the remote RADIUS server.

The screenshot shows the Local-FortiGate GUI with the navigation bar at the top. The main window displays a table titled "User Groups". The table has columns for "Group Name", "Group Type", "Members", and "Ref.". There are five entries: "Guest-group" (Firewall, guest, 0), "Remote-users" (Firewall, External\_Server, 0), "SSO\_Guest\_Users" (Fortinet Single Sign-On (FSSO), 1), and "Training" (highlighted in blue, Firewall, 0). The left sidebar shows the "User & Authentication" section selected, with "User Groups" highlighted. Other options like "Dashboard", "Network", and "Policy & Objects" are also visible.

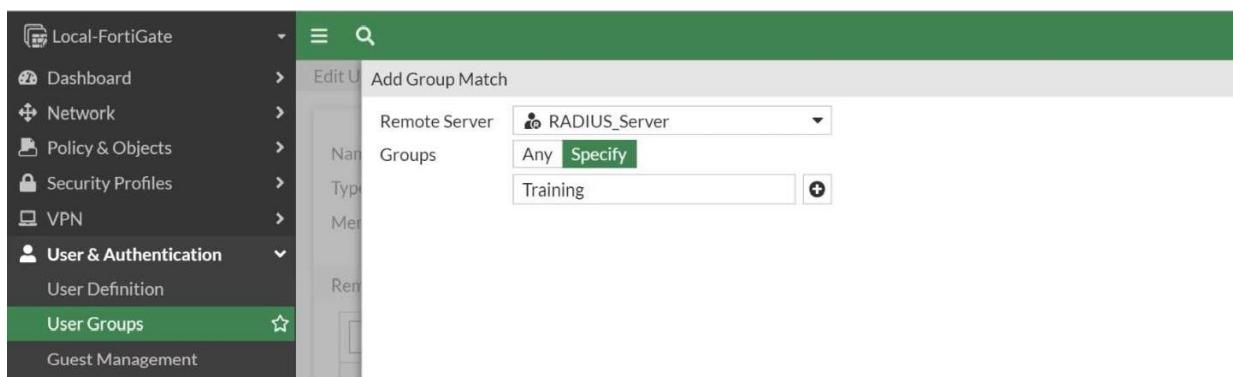
Group Name	Group Type	Members	Ref.
Guest-group	Firewall	guest	0
Remote-users	Firewall	External_Server	0
SSO_Guest_Users	Fortinet Single Sign-On (FSSO)		1
Training	Firewall		0



The **Add Group Match** window opens.



3. In the **Remote Server** field, select **RADIUS\_Server**.
4. In the **Groups** field, select **Specify**, and then type the group name **Training**.



5. Click **OK**.

The user in this RADIUS server group is now included in the FortiGate **Training** firewall user group. Only users from the remote RADIUS server that match this user group entry can authenticate.



The remote RADIUS server is configured with using the RADIUS attribute value pair (AVP) 26, known as a vendor-specific attribute (VSA). This attribute allows the Fortinet-Group-Name VSA to be included in the RADIUS response. In FortiOS, the user group must be configured to specifically match this group.

The screenshot shows the Local-FortiGate GUI with the navigation menu on the left. Under 'User & Authentication', 'User Groups' is selected. On the right, the 'Edit User Group' window is open for a group named 'Training'. The 'Type' is set to 'Firewall'. In the 'Members' section, there is a '+' button. Below that is the 'Remote Groups' section, which contains a table with one entry: 'RADIUS\_Server' under 'Remote Server' and 'Training' under 'Group Name'. There are 'Add', 'Edit', and 'Delete' buttons above the table.

6. Click **OK**.

## Add the Training User Group to the Firewall Policy

Now that you have added the RADIUS server to the **Training** firewall user group, you can add the group to a firewall policy. This allows you to control access to network resources, because policy decisions are made for the group as a whole.

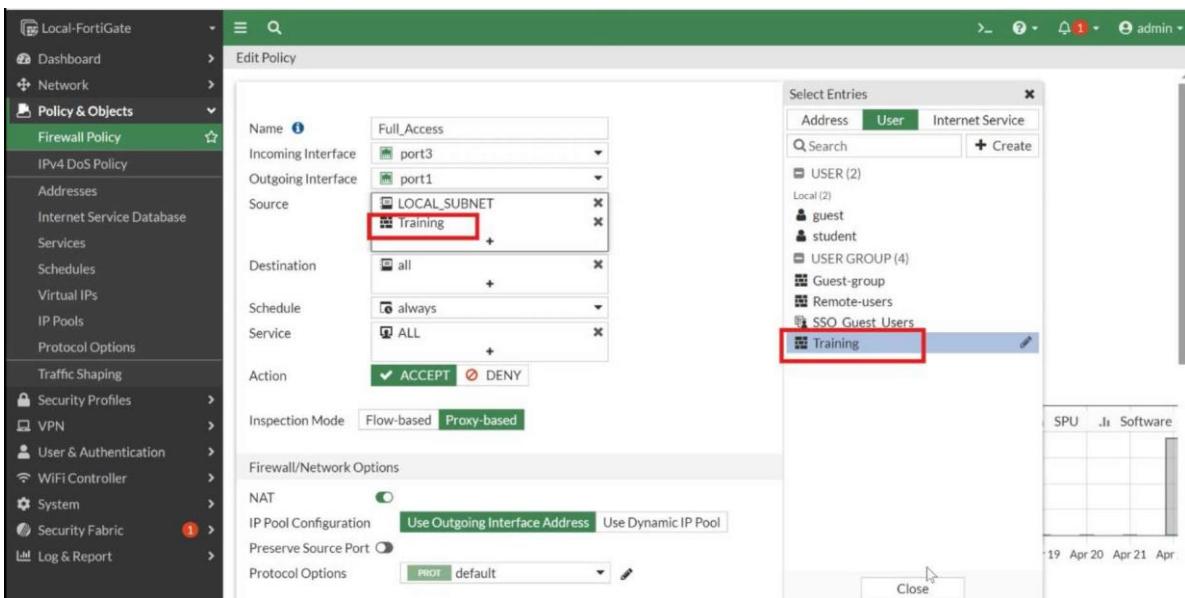
### To add the Training user group to the firewall policy

1. On the Local-FortiGate GUI, click **Policy & Objects > Firewall Policy**, and then double-click the existing port3 to port1 firewall policy.

The screenshot shows the Local-FortiGate GUI with the navigation menu on the left. Under 'Policy & Objects', 'Firewall Policy' is selected. On the right, the 'Firewall Policy' configuration window is open. It shows a single rule: 'port3 -> port1' with ID 1. The rule details are: Source is 'LOCAL\_SUBNET', Destination is 'all', Schedule is 'always', Service is 'ALL', Action is 'ACCEPT', NAT is checked, Type is 'Standard', and it uses 'WEB' and 'SSL' categories. There are 'Create new', 'Edit', and 'Delete' buttons at the top of the list.

2. Configure the following setting:

Field	Value
Source	Click <b>+</b> , and then select <b>Training</b> (located under <b>User</b> ).



### 3. Click OK.

	ID	Name	From	To	Source	Destination	Schedule	Service	Action	NAT	Type	Security Profiles
1	1	Full_Access	port3	port1	Training LOCAL_SUBNET	all	always	ALL	✓ ACCEPT	✓ NAT	Standard	WEB Category_Monitor SSL certificate-inspection
0	Implicit Deny		any	any	all	all	always	ALL	✗ DENY			

## To test whether the radius1 user can successfully authenticate

1. On the Local-FortiGate CLI, log in with the username **admin** and password **password**.
2. Enter the following command:

```
diagnose test authserver radius <RADIUS server name> mschap2
<RADIUS user name> <password>
```

Where:

- <RADIUS server name> is RADIUS\_Server (case sensitive)
- <RADIUS user name> is **radius1**
- <password> is **Training!**

A message like the following example should appear to indicate that authentication was successful:

```
Local-FortiGate #
Local-FortiGate #
Local-FortiGate # diagnose test authserver radius RADIUS_Server mschap2 radius1 Training!
authenticate 'radius1' against 'mschap2' succeeded, server=primary assigned_rad_session_id=1919812835 session_timeout=0 secs idle_timeout=0 secs!
Group membership(s) - Training
```

```
Local-FortiGate #
Local-FortiGate #
Local-FortiGate #
```

3. Close the Local-FortiGate CLI window.

# Authenticate and Monitor the Authentication

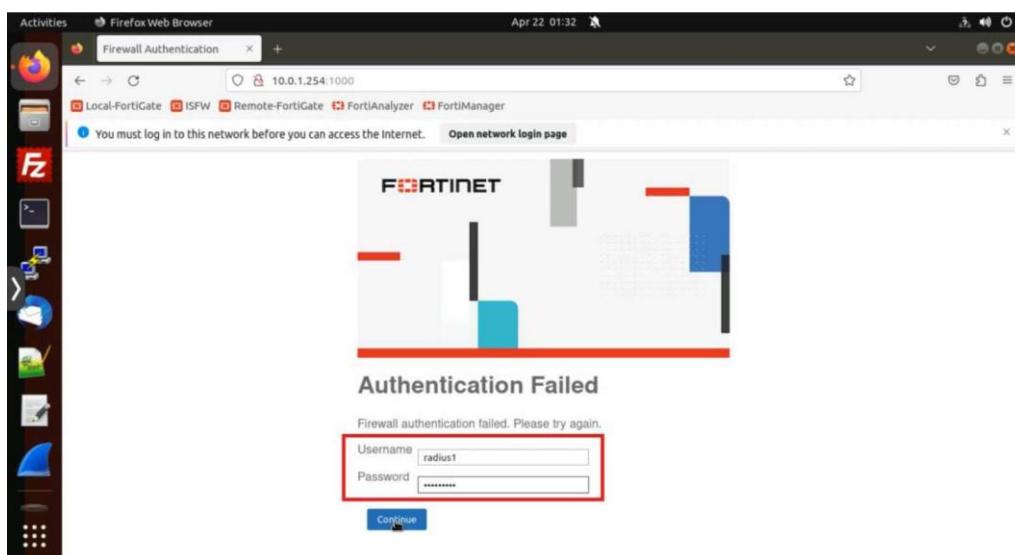
You will authenticate through the firewall policy as **radius1**. This user is a member of the **Training** group on FortiGate. Then, you will monitor the authentication.

## To authenticate as a remote RADIUS user

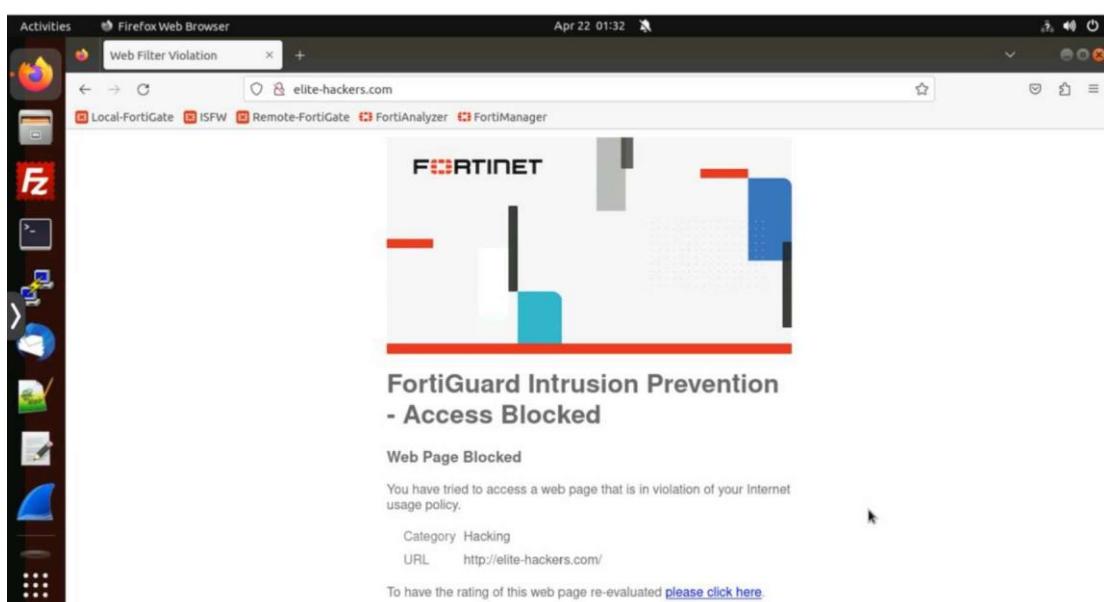
1. On the Local-Client VM, open a new browser tab, and then go to [elite-hackers.com](http://elite-hackers.com).

You are asked to log in to the network.

2. Log in as **radius1** with the password **Training!**.



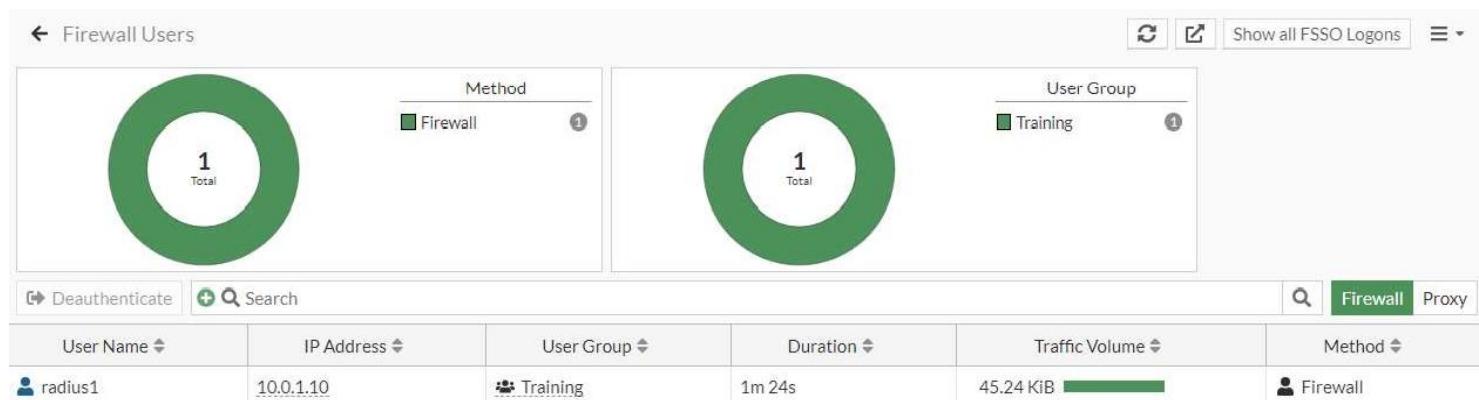
This URL is set to be blocked by the web filter security profile you enabled in the firewall policy.



Notice that the blocked page displays a **replacement message** that includes useful information, such as the **URL** and **Category**.

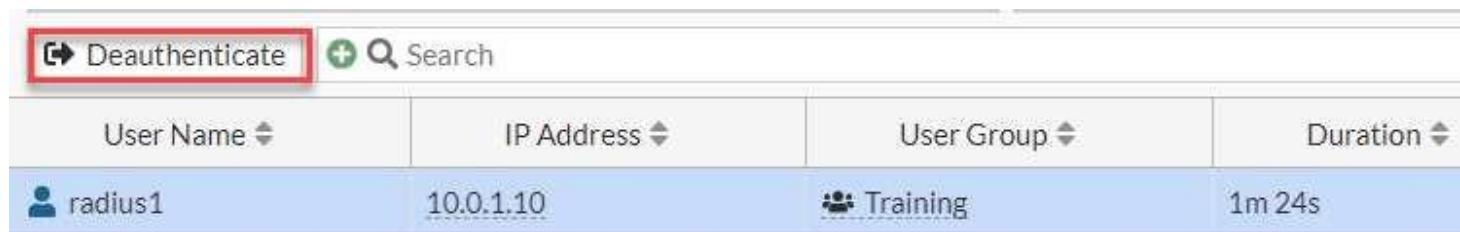
## To monitor active authenticated users

1. Return to the browser tab where you are logged in to the Local-FortiGate GUI as **admin**.
2. Click **Dashboard > Assets&Identities**, and then click **Firewall Users** to expand it to full screen to view this login authentication and monitor the firewall authenticated user.



You will see the user **radius1** listed along with other information, such as **User Group** and **IP Address**.

3. Click **aduser1**, and then click **Deauthenticate**.



The **config user setting** CLI command determines how long a user can remain authenticated. However, you can choose to manually revoke a user authentication by selecting the user in the **Firewall User Monitor** list, and then clicking **Deauthenticate**. After the user is deauthenticated, the user disappears from the list, because it is reserved for active users only.

4. In the **Confirm** window, click **OK**.



This deauthenticates the user. The user must log in again to access the resources that the firewall policy protects.

# Remove the User Group from the Firewall Policy

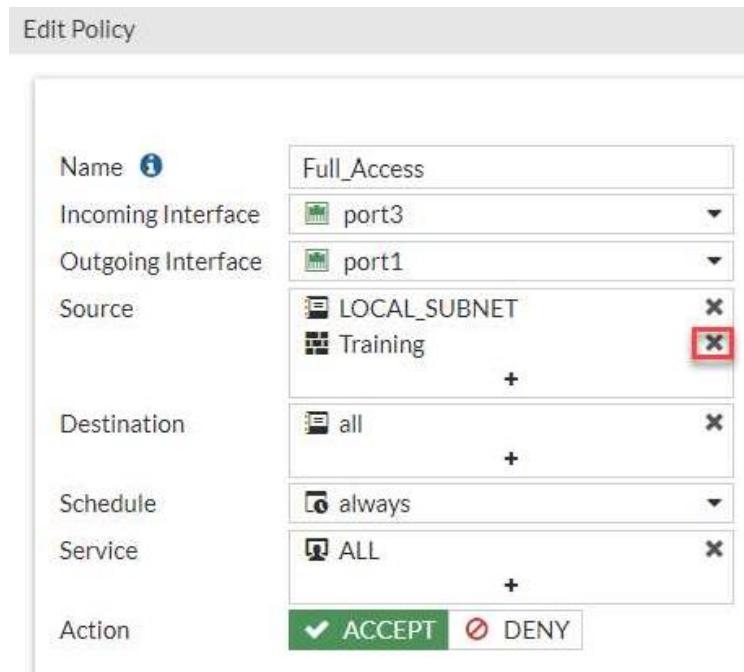
You will remove the user group assigned to the firewall policy for authentication.

## To remove the remote user group from the firewall policy

1. On the Local-FortiGate GUI, click **Policy & Objects > Firewall Policy**, and then double-click the existing port3 to port1 firewall policy.

	ID	Name	From	To	Source	Destination	Schedule	Service	Action	IP Pool
<input type="checkbox"/>	1	Full_Access	port3	port1	Training LOCAL_SUBNET	all	always	ALL	ACCEPT	
<input type="checkbox"/>	0	Implicit Deny	<input type="checkbox"/> any	<input type="checkbox"/> any	all	all	always	ALL	DENY	

2. In the **Source** field, remove the **Training** user group.



3. Click **Close**, and then click **OK** to save the changes.