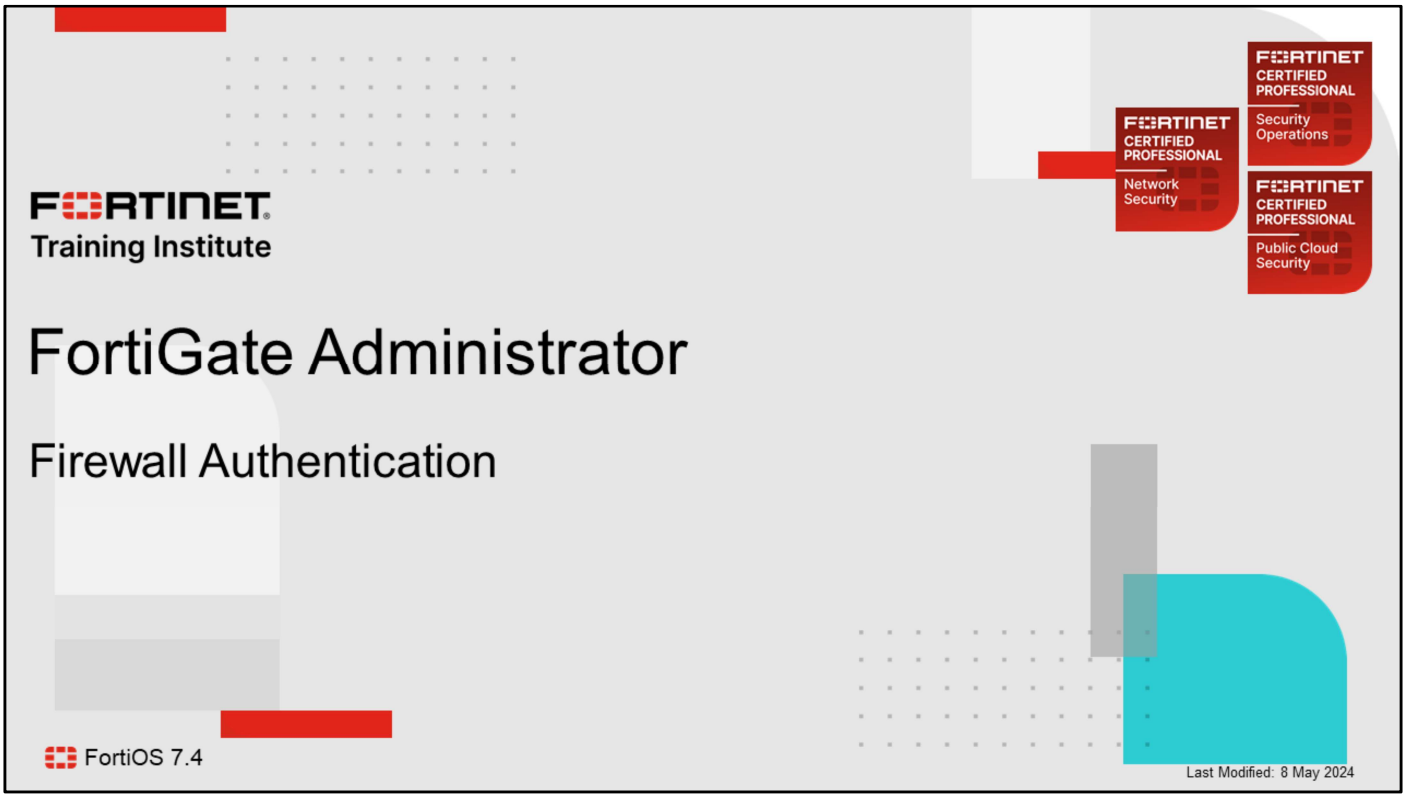


In this lesson, you will learn about using authentication on the firewall policies of FortiGate.

Objectives

- Configure a remote LDAP authentication server on FortiGate
- Configure a remote RADIUS authentication server on FortiGate
- Deploy active and passive authentication
- Monitor firewall users using the FortiGate GUI

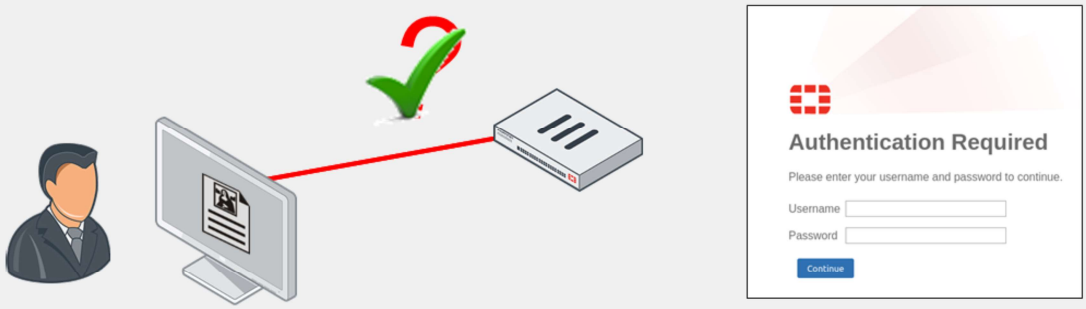
FORTINET Training Institute

After completing this section, you should be able to achieve the objectives shown on this slide.

By demonstrating competence in methods of firewall authentication, you will be able to describe and identify the supported methods of firewall authentication available on FortiGate.

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

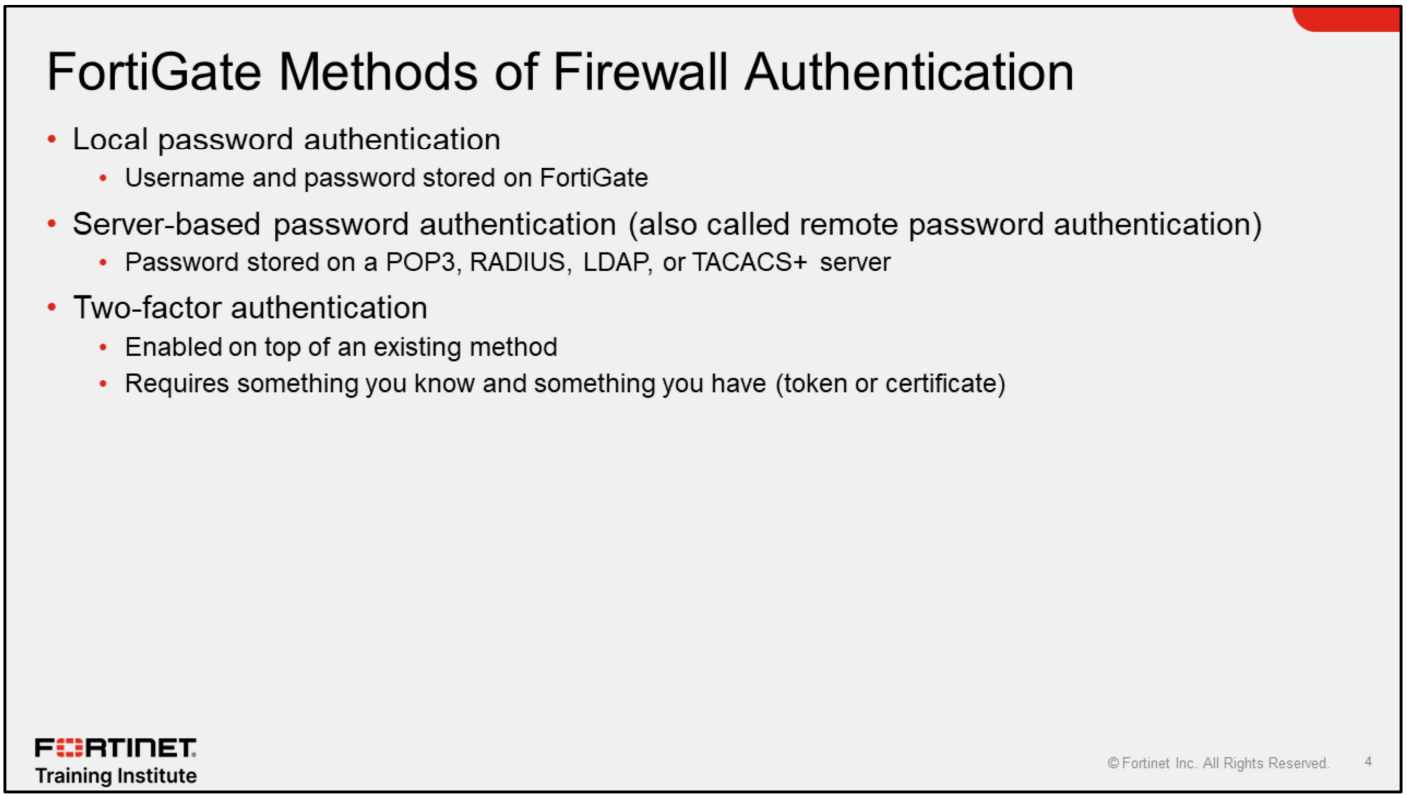


FORTINET Training Institute

Traditional firewalling grants network access by verifying the source IP address and device. 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.

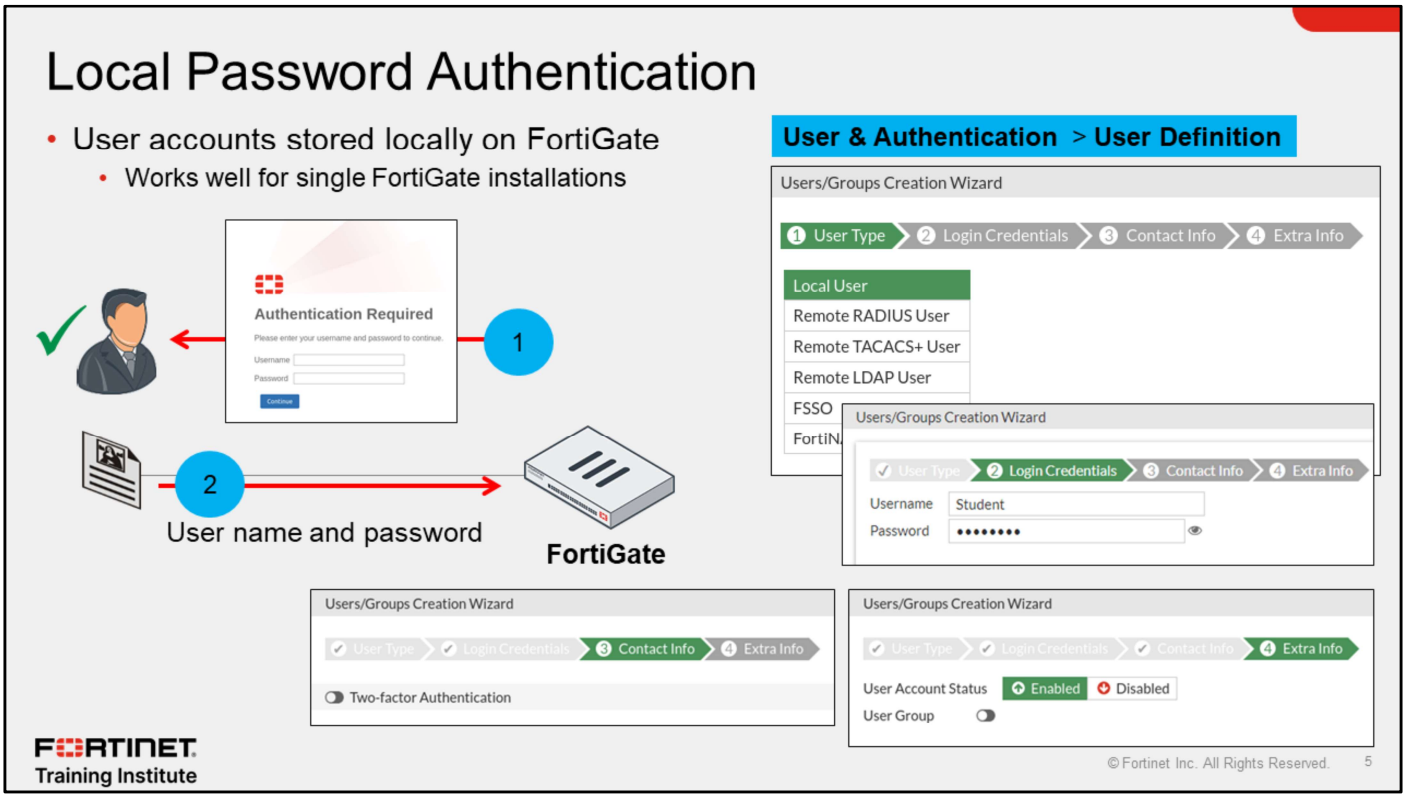
Where access is controlled by a user or user group, users must authenticate by entering valid credentials (such as username and password). After FortiGate validates the user, FortiGate applies firewall policies and profiles to allow or deny access to specific network resources.



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 lesson, you will learn about each method of firewall authentication in detail.



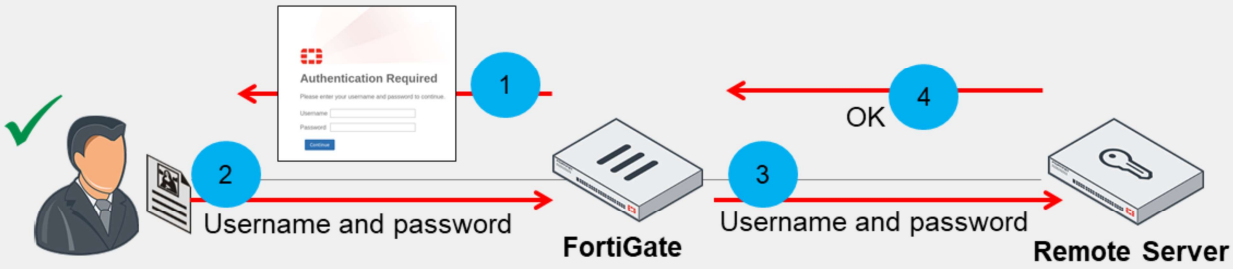
The simplest method of authentication is local password authentication. User account information (username and password) is stored locally on the FortiGate device. This method works well for a single FortiGate installation.

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 preconfigured 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.

Server-Based Password Authentication

- Accounts are stored on a remote authentication server
- Administrators can do one of the following:
  - Create an account for the user locally, and specify the server to verify the password
  - Add the authentication server to a user group
    - All users in that server become members of the group

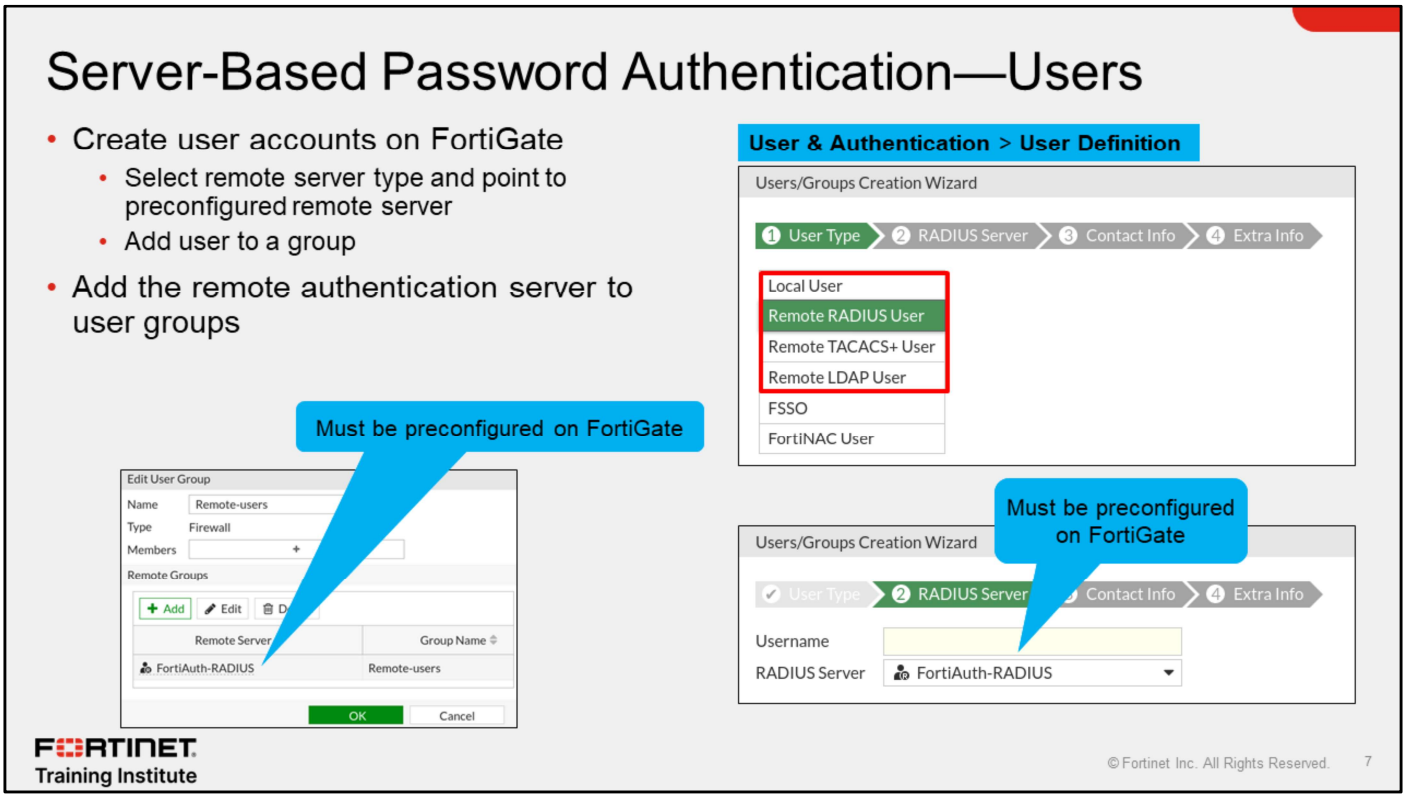


FORTINET Training Institute

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.



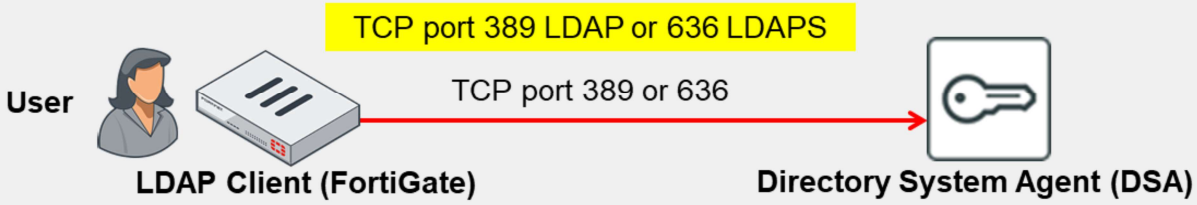
You can configure FortiGate to use external authentication servers in the following two ways:

- 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 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

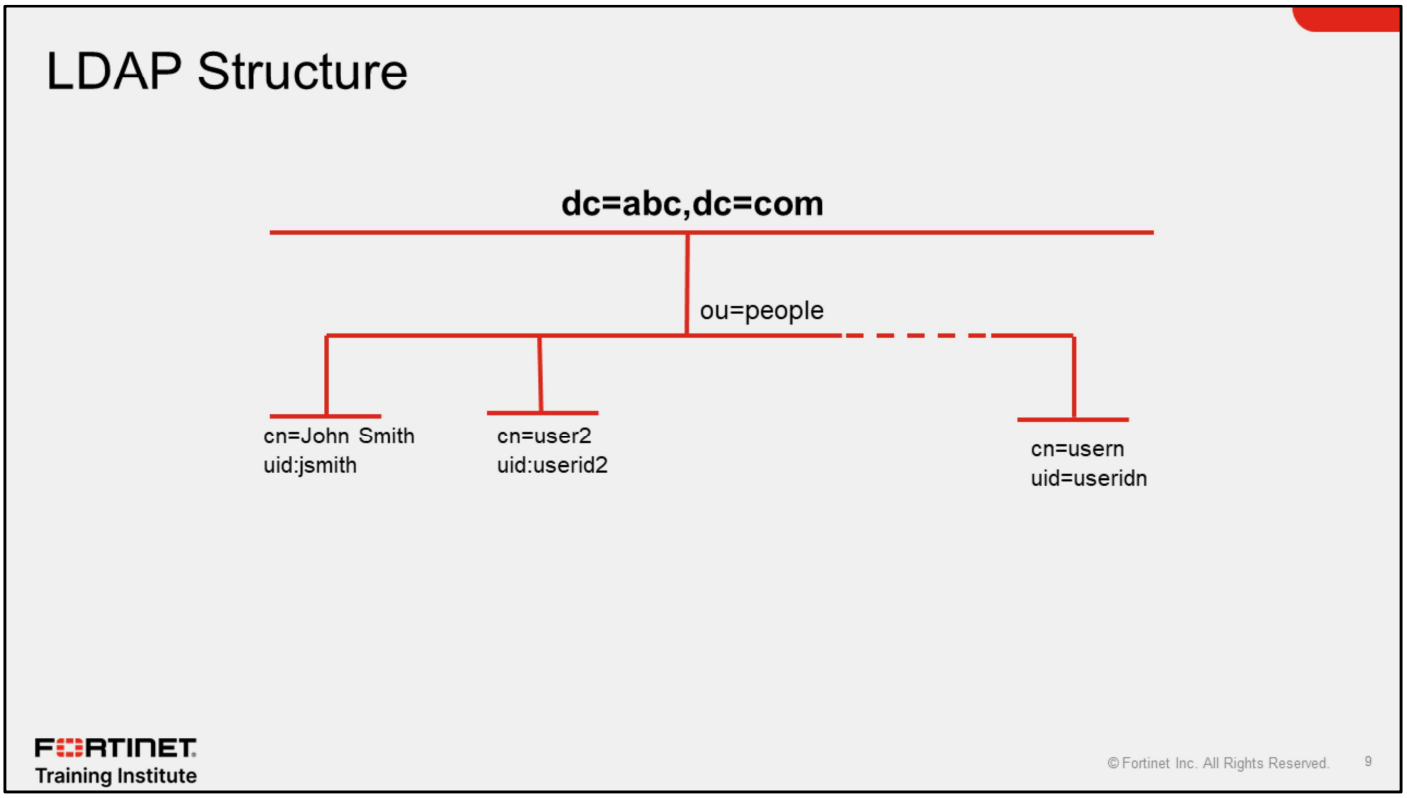
FORTINET Training Institute

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

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 search, compare, and add or delete 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.

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 which is more secure.

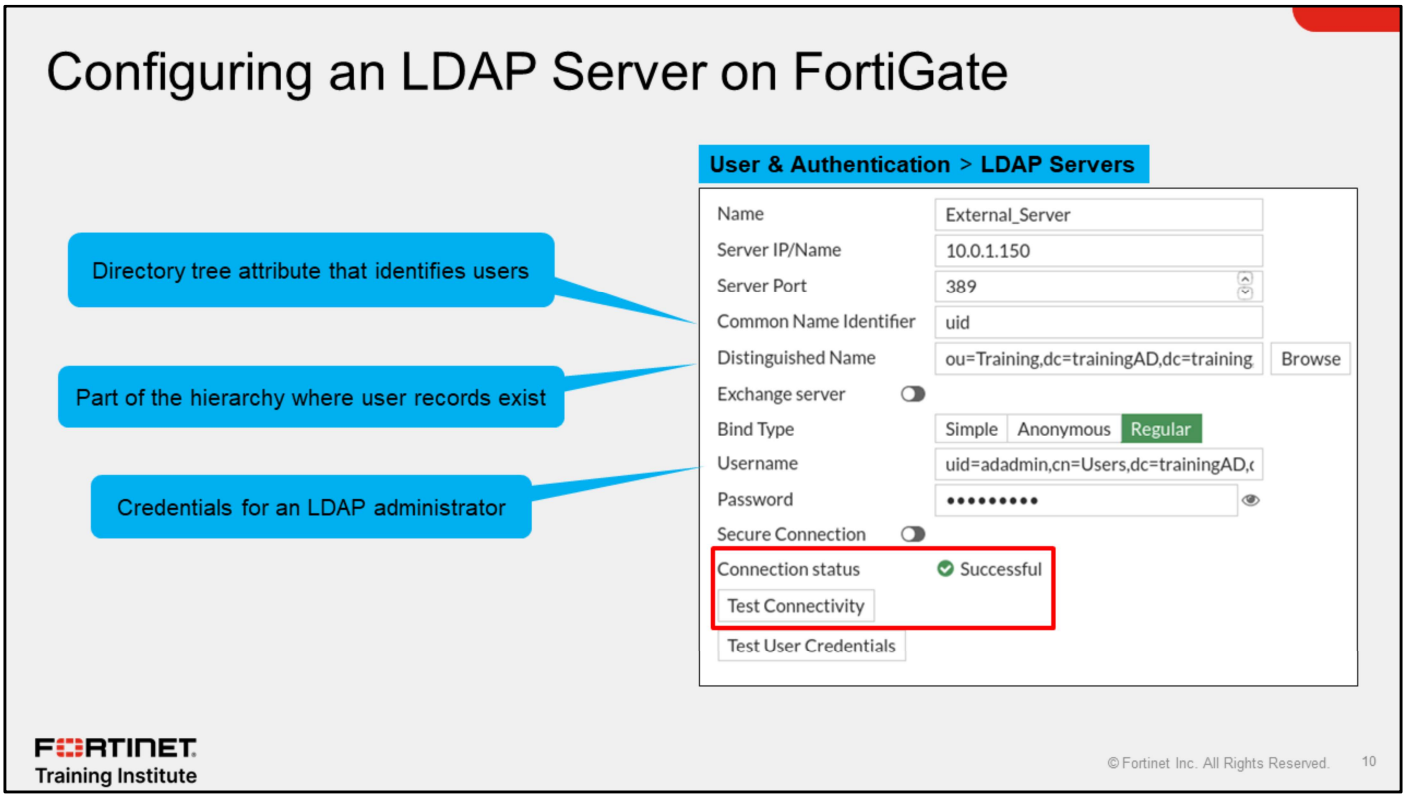


The LDAP structure is similar to a tree that contains entries (objects) in each branch. An LDAP server hierarchy often reflects the hierarchy of the organization it serves. The root represents the organization itself, usually defined as domain component (DC), and a DNS domain, such as abc.com (because 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.



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.

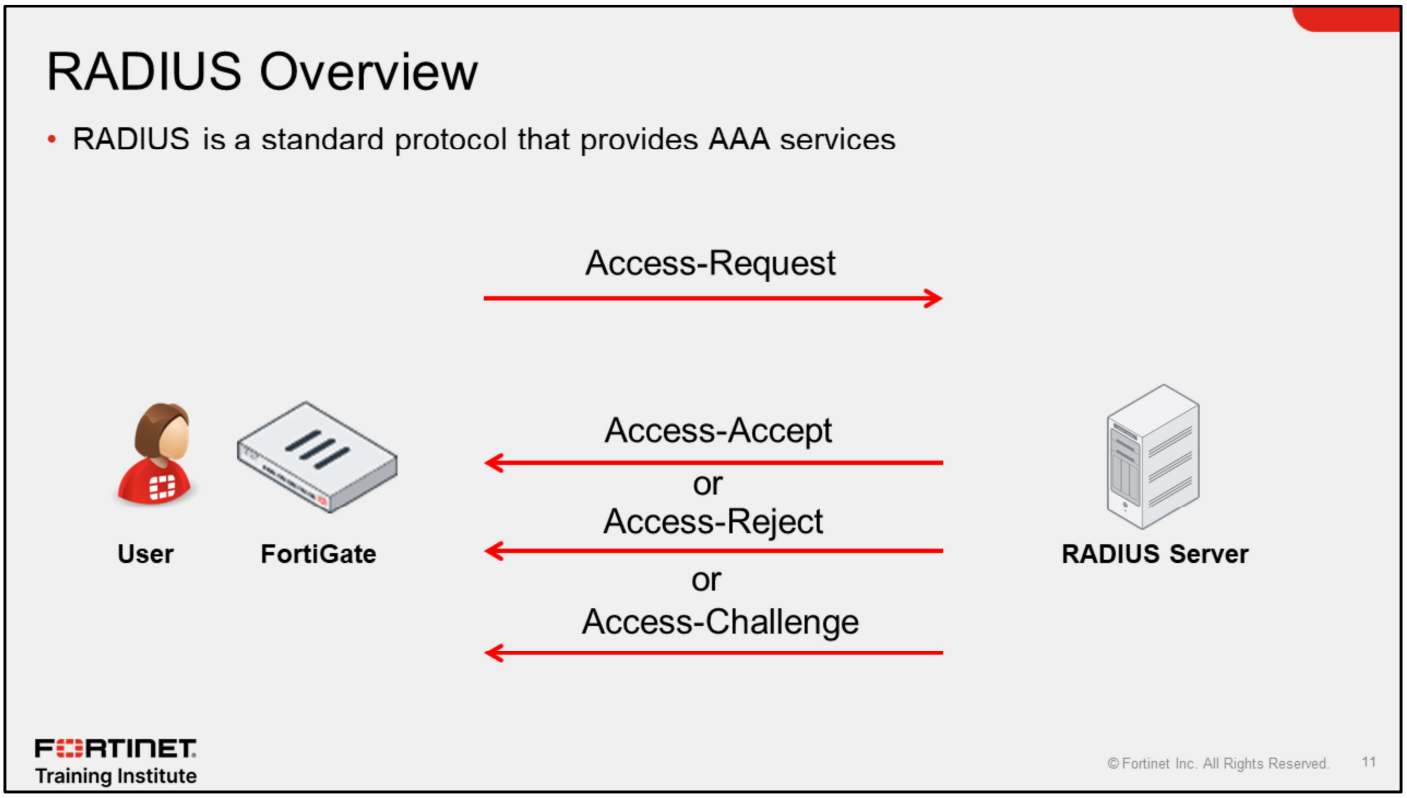
The **Common Name Identifier** setting is the attribute name you use to find the user name. Some schemas allow you to use the attribute userid. AD most commonly uses `sAMAccountName` or `cn`, but can use others as well.

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.

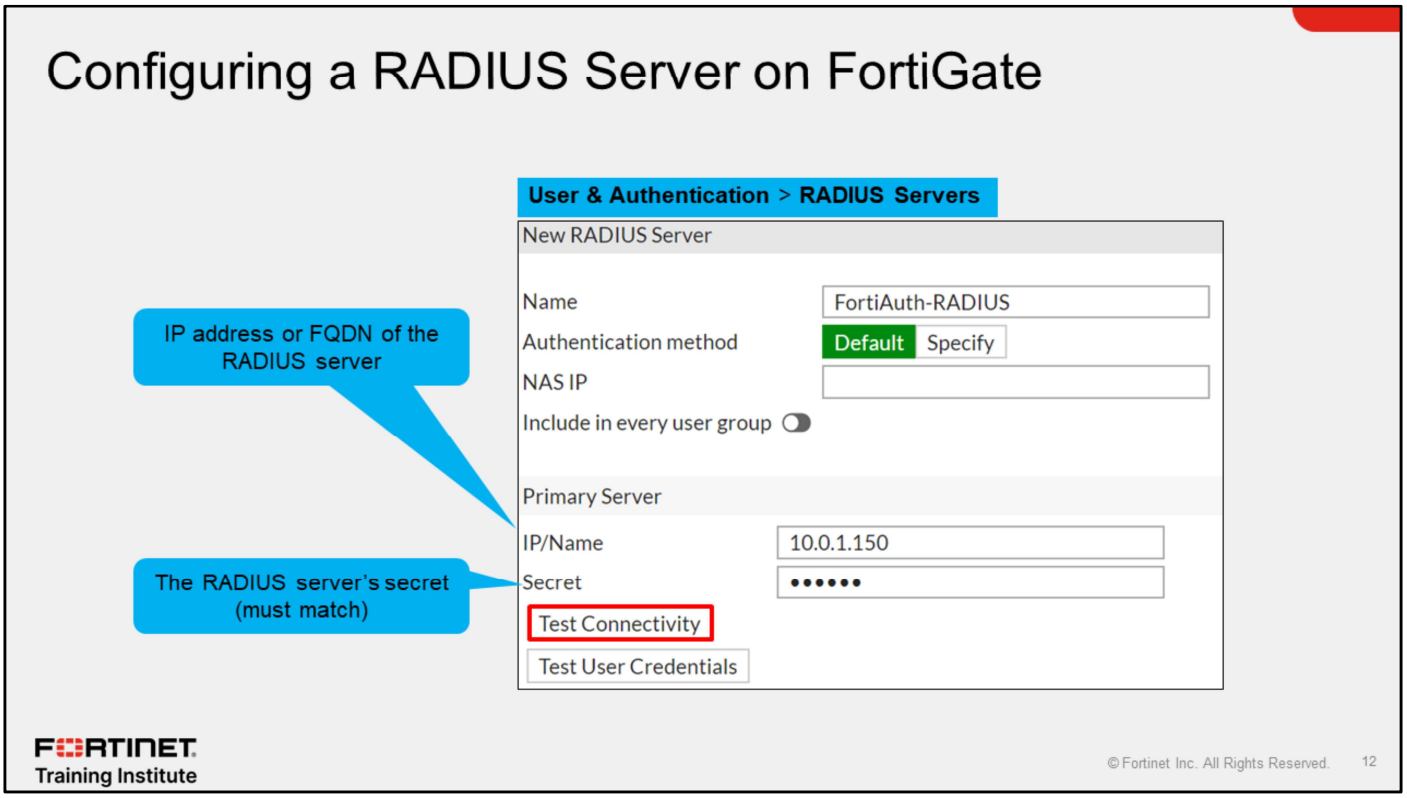


RADIUS is much different from LDAP, because there is no directory tree structure to consider. RADIUS is a standard protocol that provides authentication, authorization, and accounting (AAA) services.

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.



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

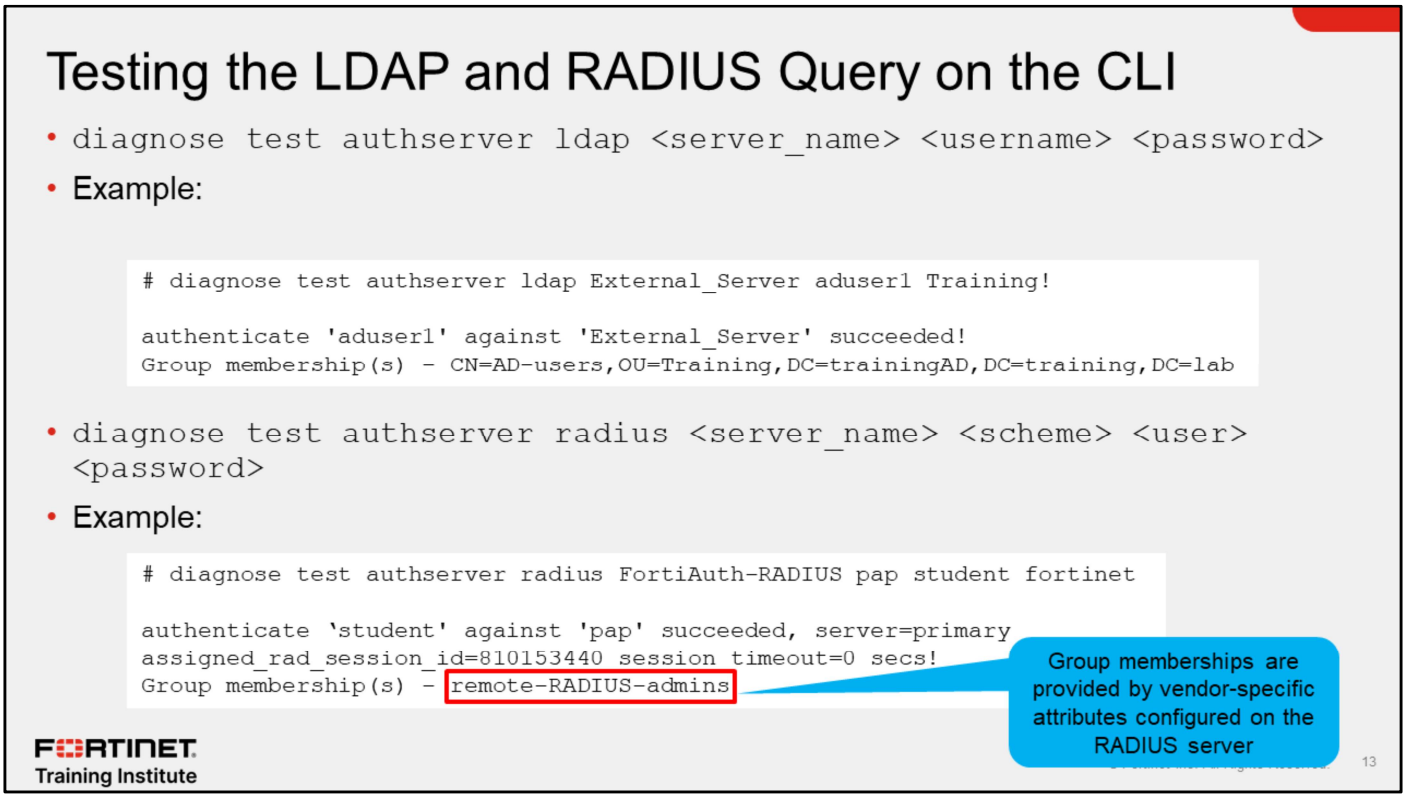
The **Primary Server IP/Name** setting is the IP address or FQDN of the RADIUS server.

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).



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

- 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

FORTINET Training Institute

Traditional user authentication requires your user name 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.

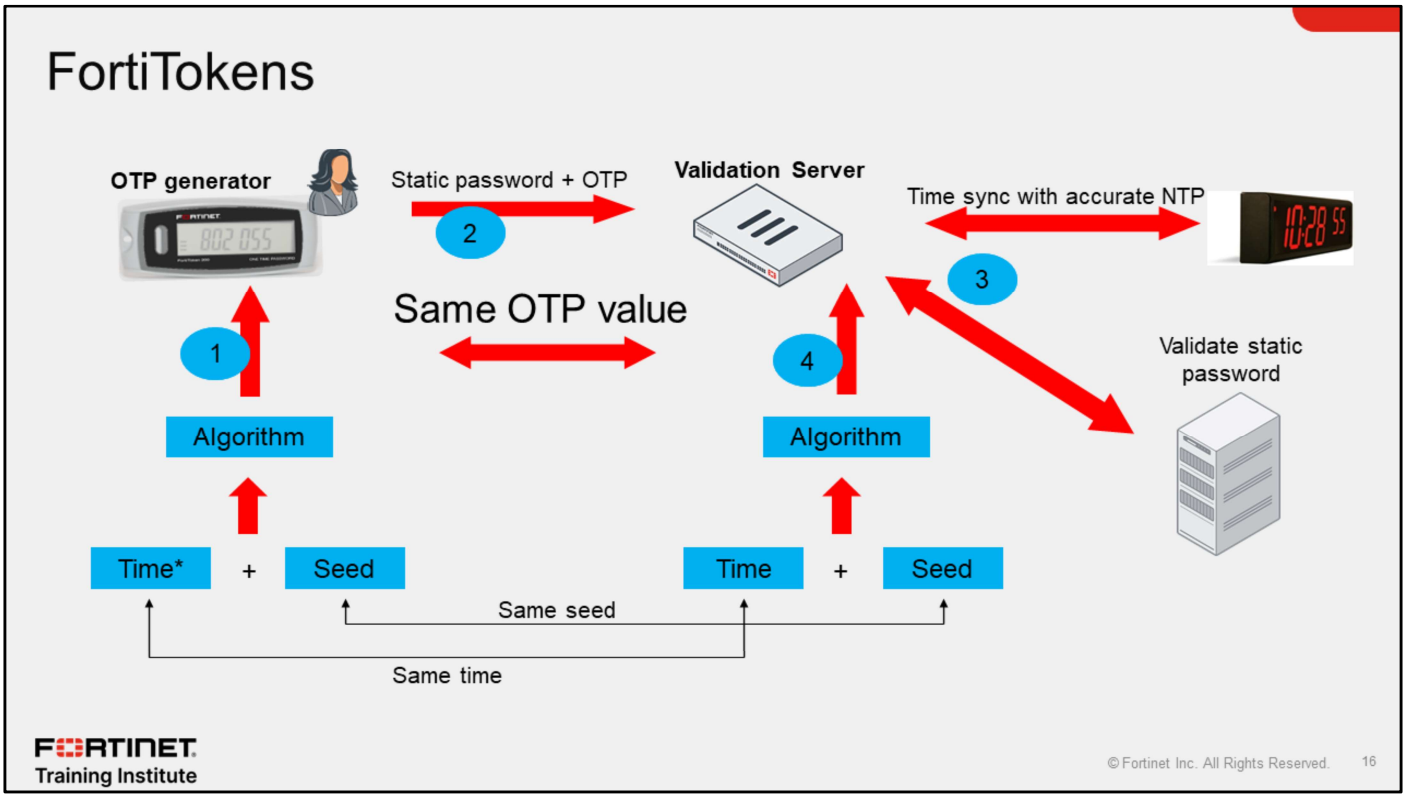
Two-Factor Authentication (Contd)

- 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!

FORTINET Training Institute

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. 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.

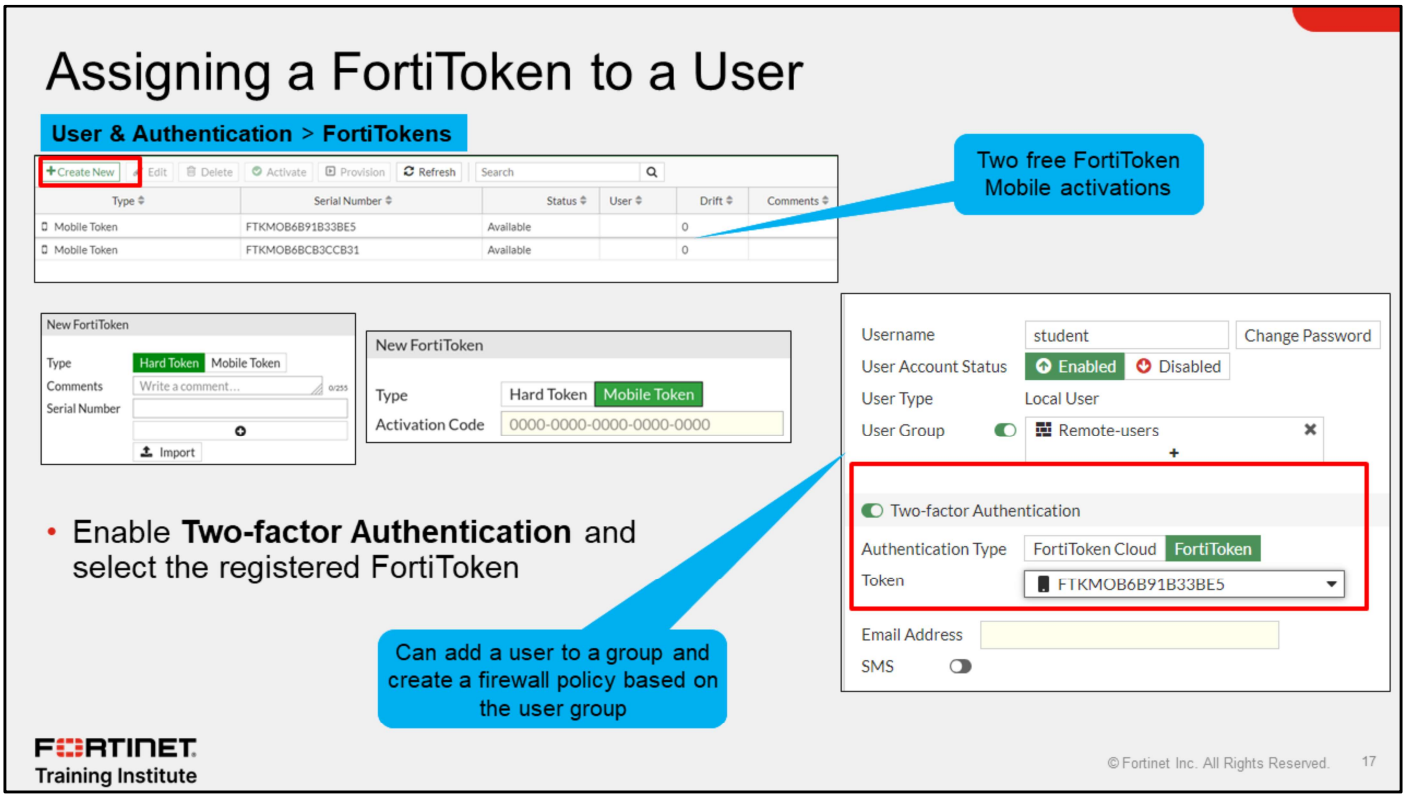


Tokens use a specific algorithm to generate an OTP. The algorithm consists of:

- 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).



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.

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

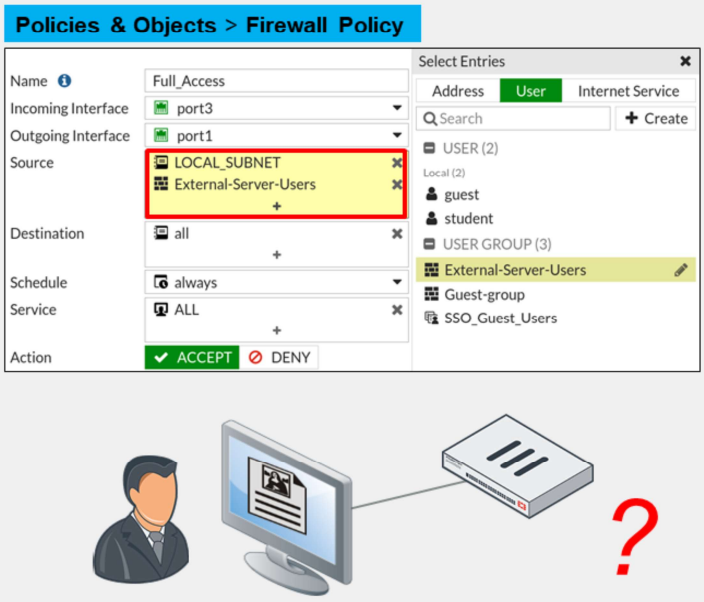
FORTINET Training Institute

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

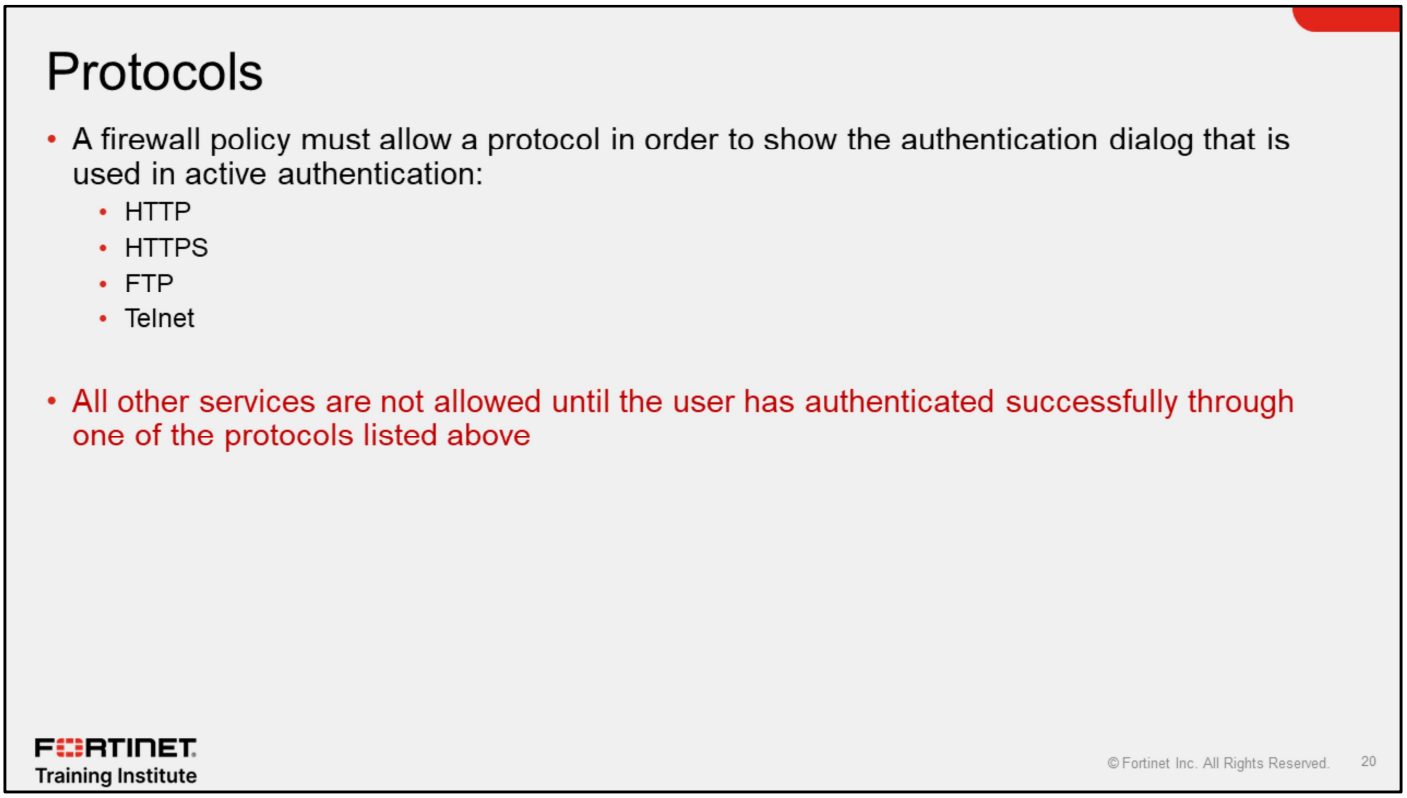
- Firewall policies can use user and user group objects to define the source. The objects include:
  - Local firewall accounts
  - External (remote) server accounts
  - PKI (certificate) users
  - FSSO users
- Anyone who belongs to the group and provides correct information will have a successful authentication



FORTINET Training Institute

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.



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

- 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

FORTINET Training Institute

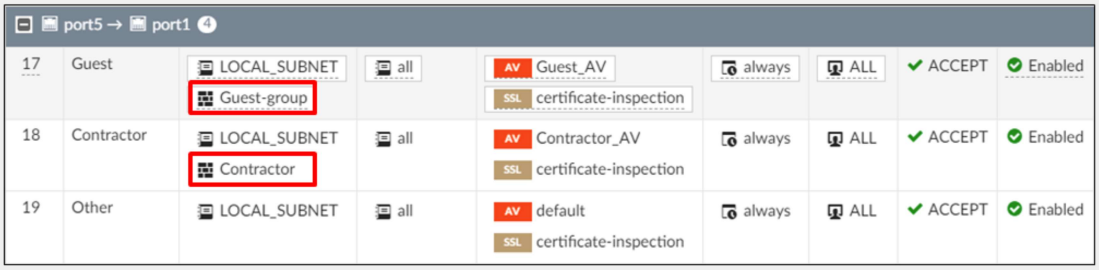
A firewall policy also checks the service in order to transport the named protocols or group of protocols. No service (with the exception of DNS) is allowed through the firewall policy before successful user authentication. DNS is usually used by HTTP so that people can use domain names for websites, 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.

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

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



- 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

FORTINET Training Institute

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.

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.

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

- 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

FORTINET Training Institute

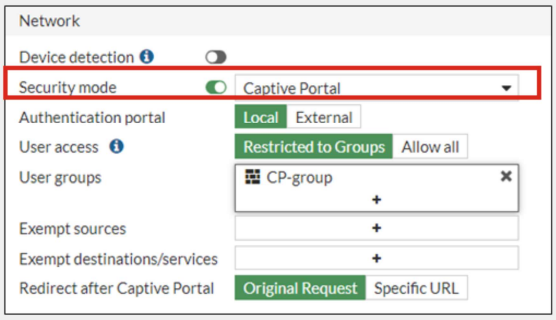
As mentioned earlier, there are three different ways you can alter active authentication behavior. If you have an active authentication firewall policy followed by a fall-through policy that does not have authentication enabled on it, then all traffic will use the fall-through policy. This means that 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.

Active Authentication Behavior (Contd)

- 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



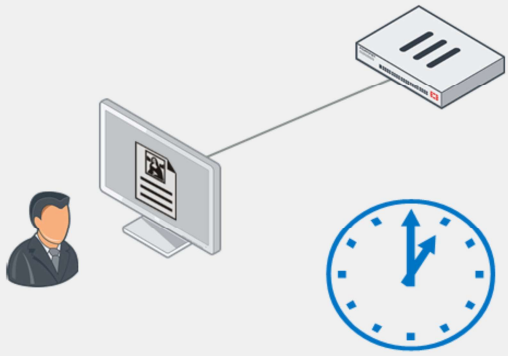
FORTINET Training Institute

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.

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



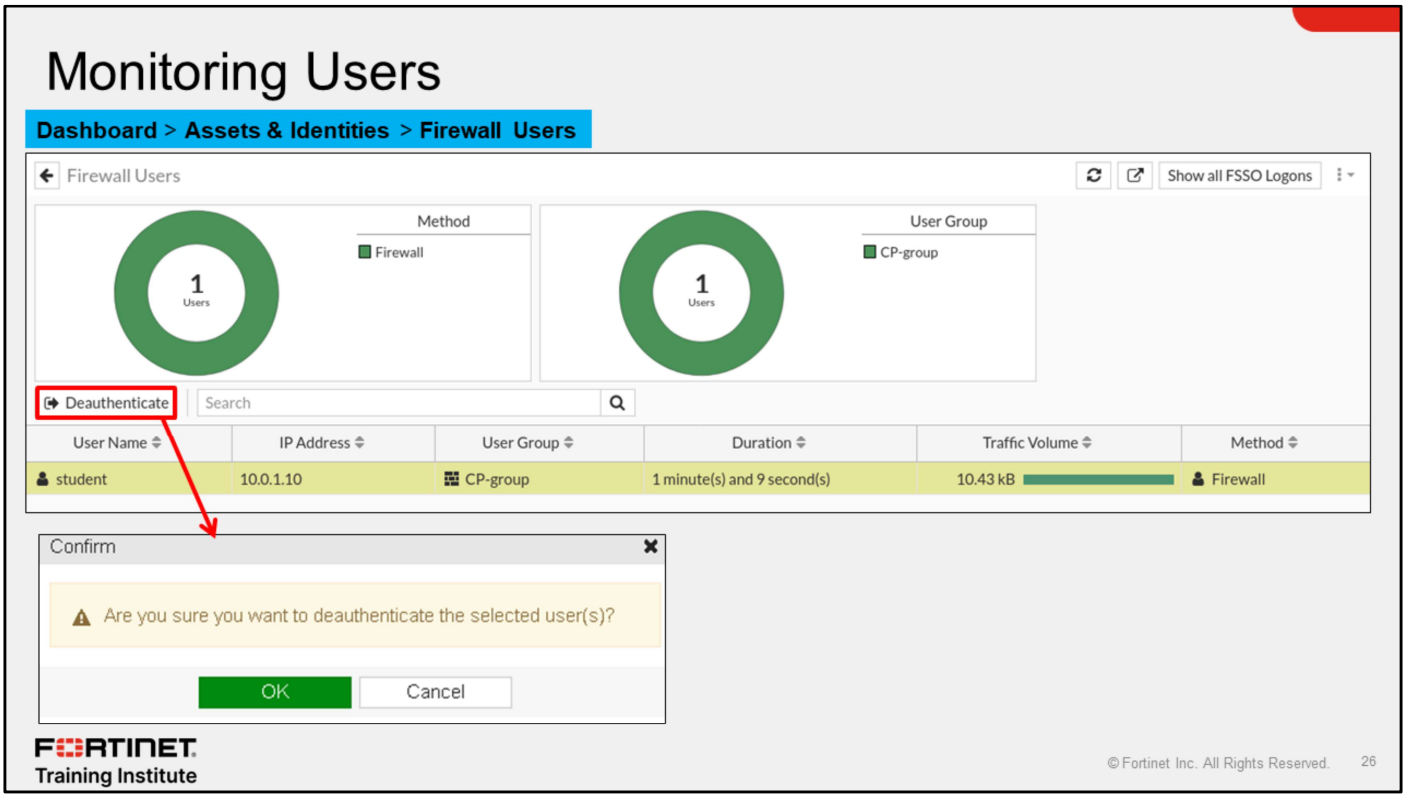
FORTINET Training Institute

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**: This looks at the packets from the host IP. If there are no packets generated by the host device in the configured timeframe, then the user is logged out.
- **Hard**: Time is an absolute value. Regardless of the user's behavior, the timer starts as soon as the user authenticates and expires after the configured value.
- **New session**: Even if traffic is being generated on existing communications channels, the authentication expires if no new sessions are created through the firewall from the host device within the configured timeout value.

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



You can monitor users who authenticate through your firewall policies using the **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.

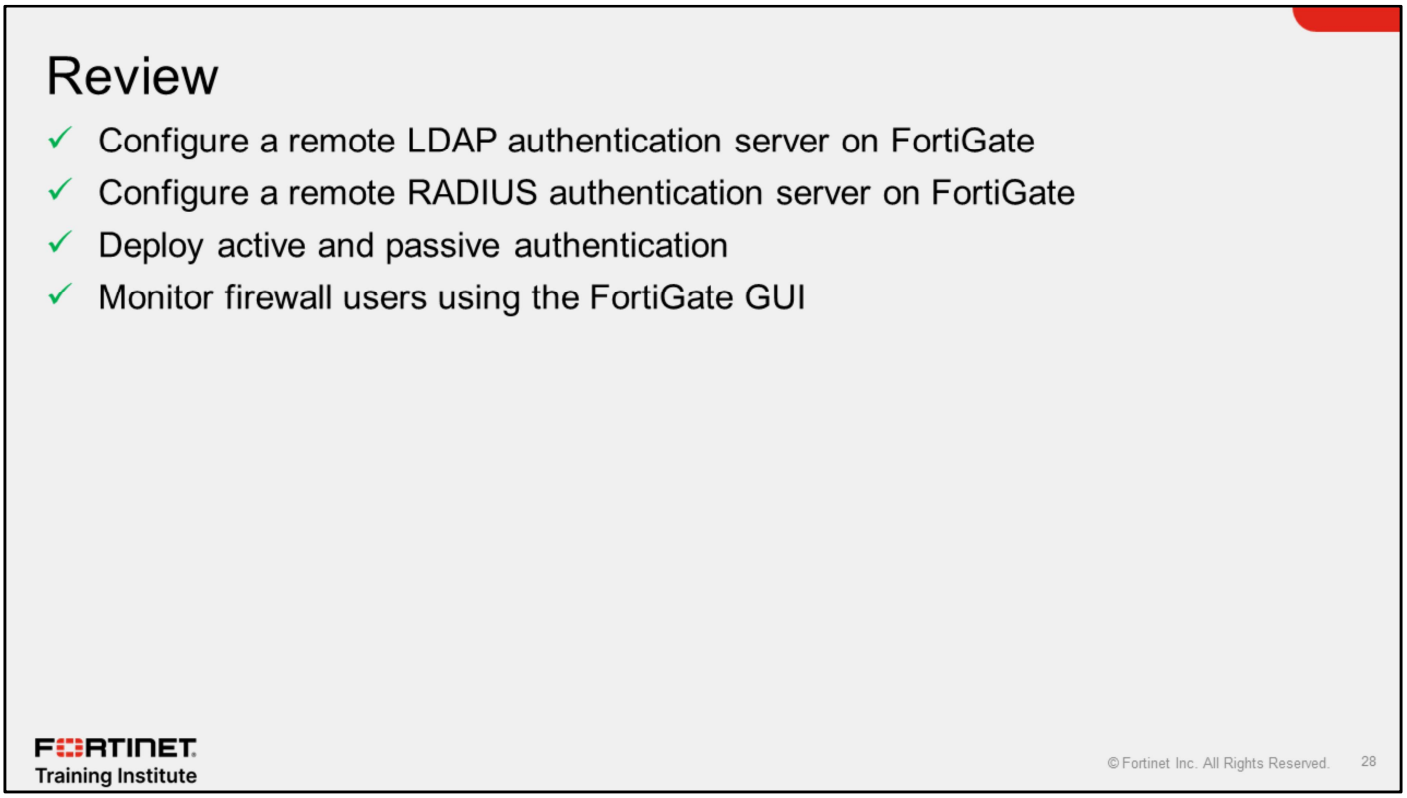
This page also allows you to disconnect a user, or multiple users, at the same time.

Knowledge Check

1. A remote LDAP user is trying to authenticate with a username and password. How does FortiGate verify the login credentials?
   - A. FortiGate queries its own database for user credentials.
   - ✓ B. FortiGate sends the user-entered credentials to the remote server for verification.

2. When FortiGate uses a RADIUS server for remote authentication, which statement about RADIUS is true?
   - A. FortiGate must query the remote RADIUS server using the distinguished name (dn).
   - ✓ B. RADIUS group memberships are provided by vendor-specific attributes (VSAs) configured on the RADIUS server

3. Which statement about active authentication is true?
   - A. Active authentication is always used before passive authentication.
   - ✓ B. The firewall policy must allow the HTTP, HTTPS, FTP, and/or Telnet protocols in order for the user to be prompted for credentials.



This slide shows the objectives that you covered in this lesson.

By mastering the objectives covered in this lesson, you learned how to use authentication on the firewall policies of FortiGate.