NFV 511-WBT vRouter Management & Logging

NFV 511-WBT AT&T Vyatta 5600 vRouter Management & Logging

The training materials and other content provided herein for assistance in training on the Vyatta vRouter may have references to Brocade as the Vyatta vRouter was formerly a Brocade product prior to AT&T's acquisition of Vyatta. Brocade remains a separate company and is not affiliated to AT&T.

AT&T

Welcome to the AT&T vRouter Dynamic Multipoint VPN course.

NFV 511-WBT vRouter Management & Logging

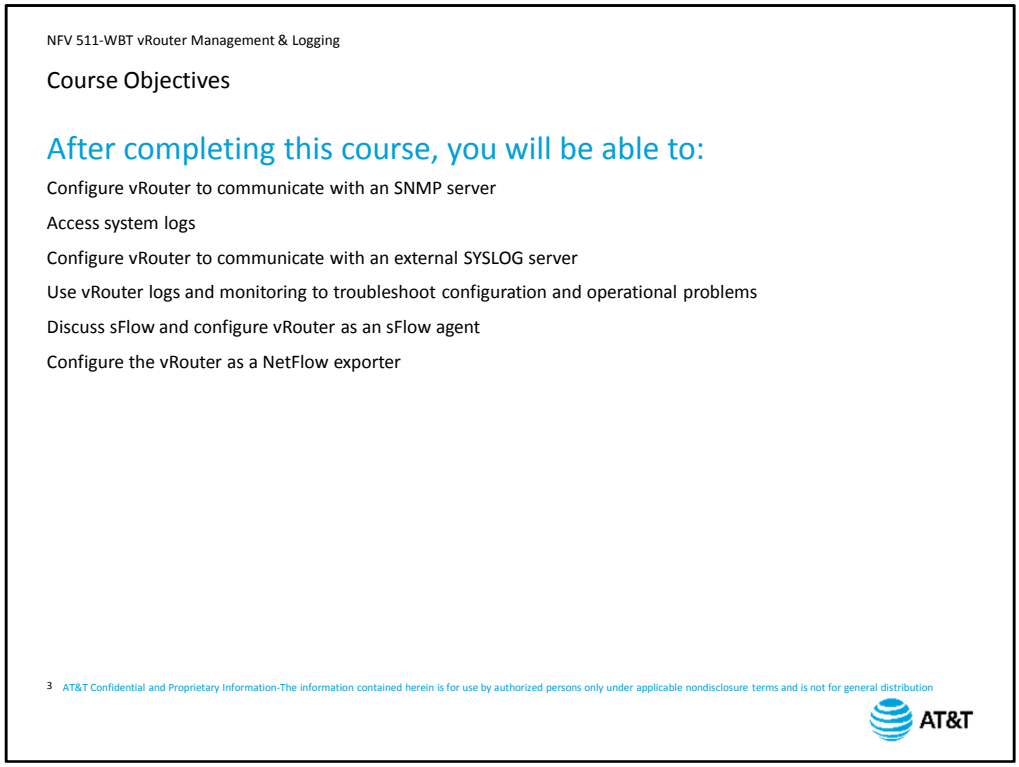
Legal Disclaimer

All or some of the products detailed in this presentation may still be under development and certain specifications may be subject to change. Nothing in this presentation shall be deemed to create a warranty of any kind.

AT&T

Before we begin the course, please take a moment to read our legal disclaimer.

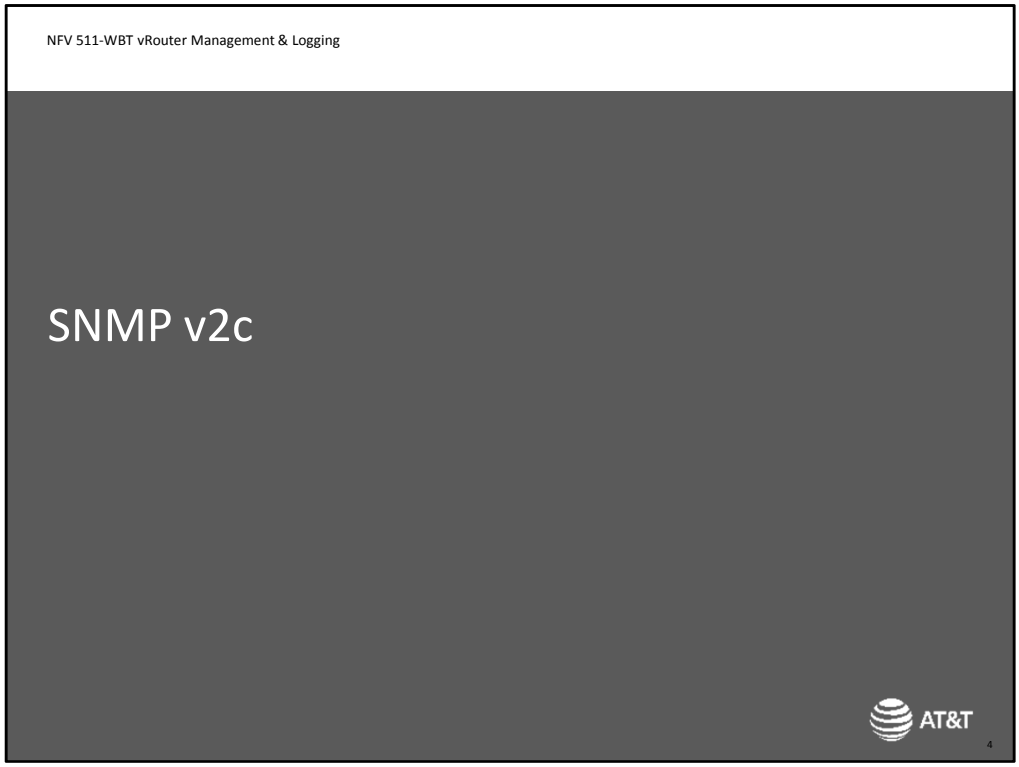
AT&T Vyatta 5600 vRouter Management and Logging



After completing this course, you should be able to:

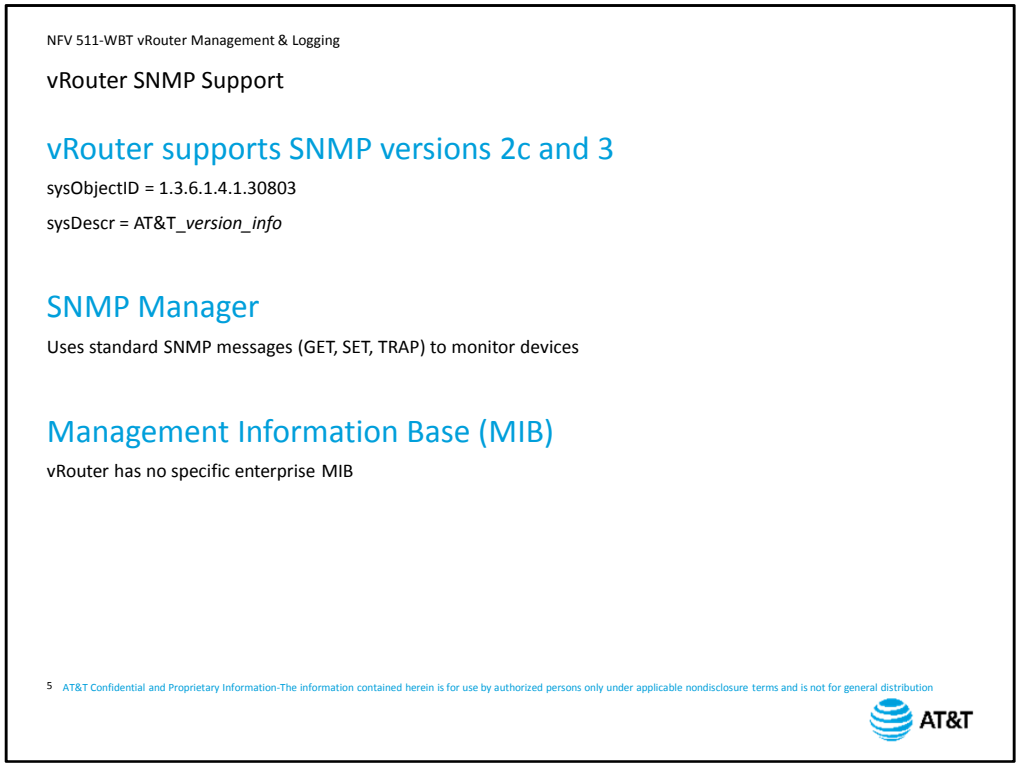
- Configure the vRouter to communicate with a SNMP server
- Access system logs
- Configure the vRouter to communicate with an external Syslog server
- Use vRouter logs and monitoring to troubleshoot configuration and operational problems
- Discuss sFlow and configure vRouter as an sFlow agent
- Configure the vRouter as a NetFlow exporter

AT&T Vyatta 5600 vRouter Management and Logging

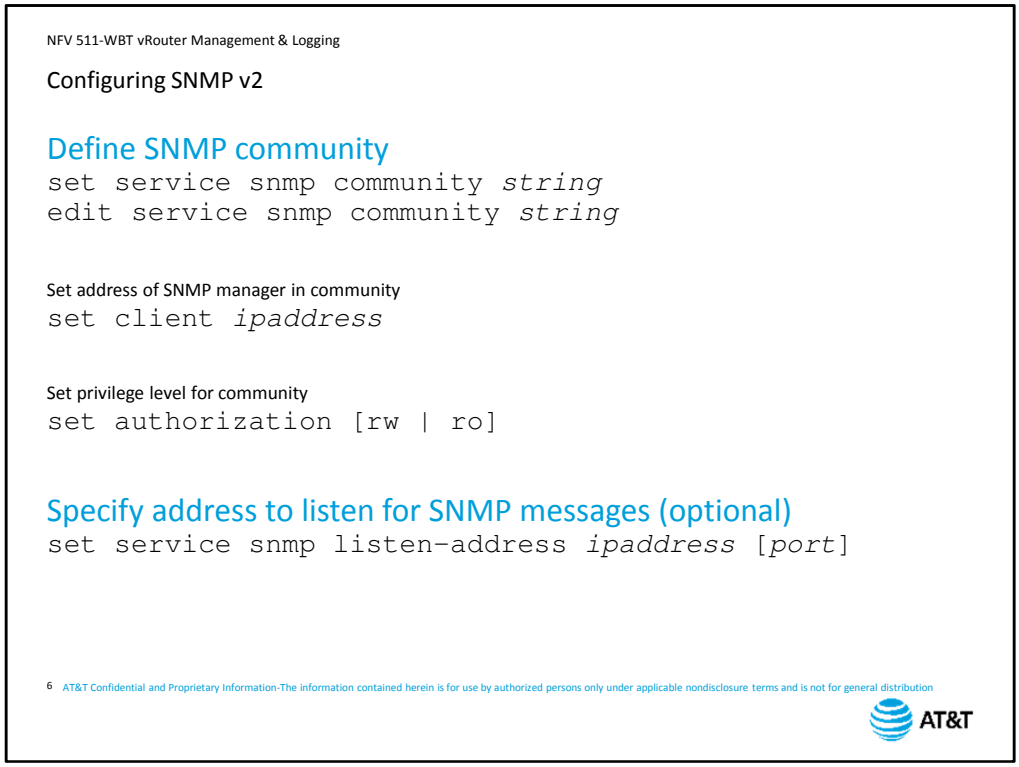


We'll begin with an overview of 5600 vRouter SNMP support, as well as the steps for configuring SNMP.

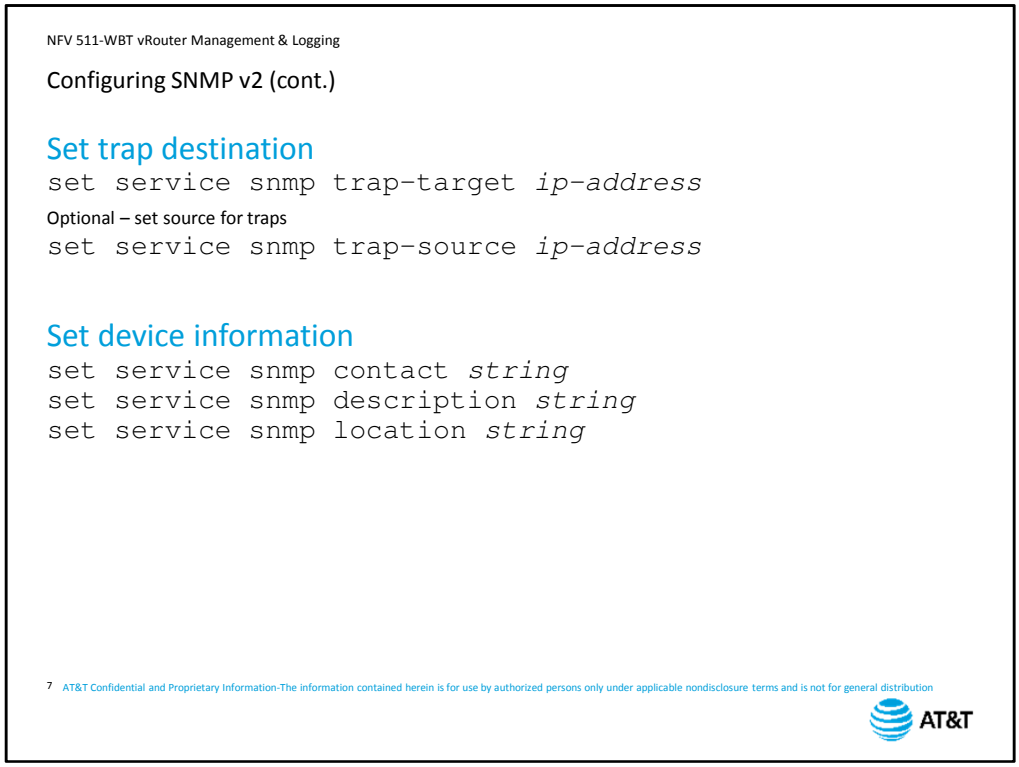
AT&T Vyatta 5600 vRouter Management and Logging



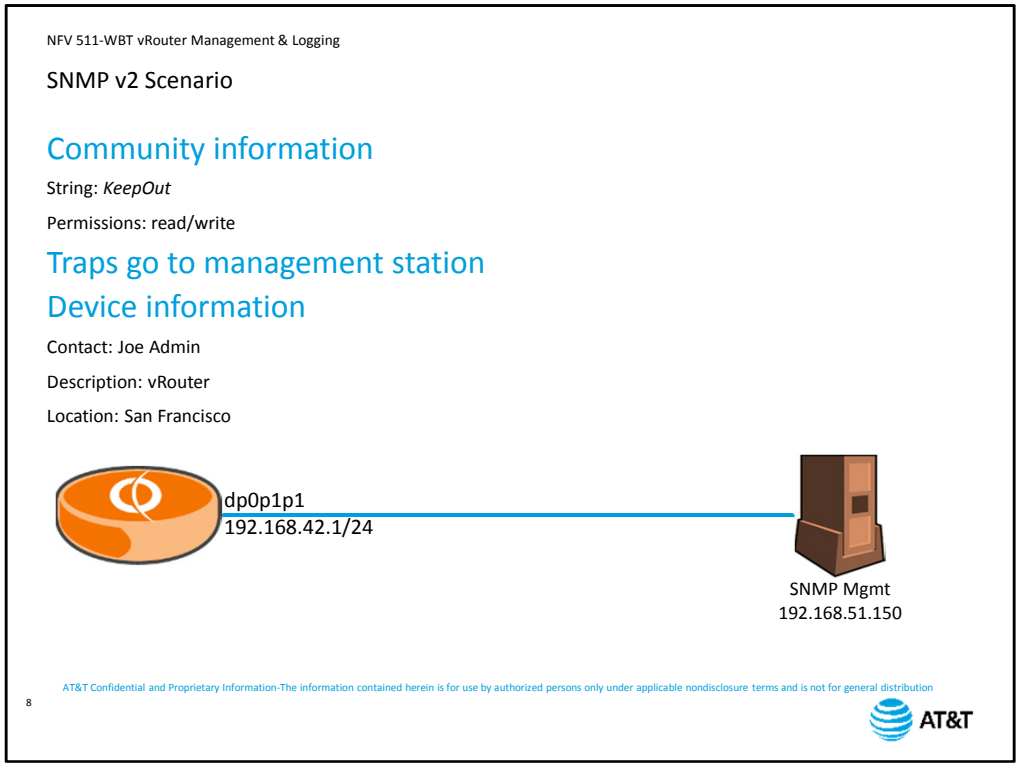
Simple Network Management Protocol (SNMP), is a standard for managing and monitoring network devices, including vRouters. The vRouter supports both SNMP version 2c and 3. We will discuss configuration for version 2c first.
The vRouter is pre-configured with the sysObjectID and sysDescr string as shown on the screen.
This system information can be pulled into your SNMP management station using standard SNMP GET messages. GET messages can also gather device operational statistics. You can use SET commands to set event conditions and thresholds on the vRouter. When the event occurs, or when the vRouter reaches a threshold, it will send a TRAP to the management station.
The actual thresholds and events, as well as monitoring settings, are defined in a Management Information Base (MIB).
Unlike other vendors, vRouter does not have a customized MIB specific to a vRouter. Instead, we utilize industry-standard MIBs. The vRouter support team has provided a list of MIBs that they use in production and lab environments to monitor vRouters.



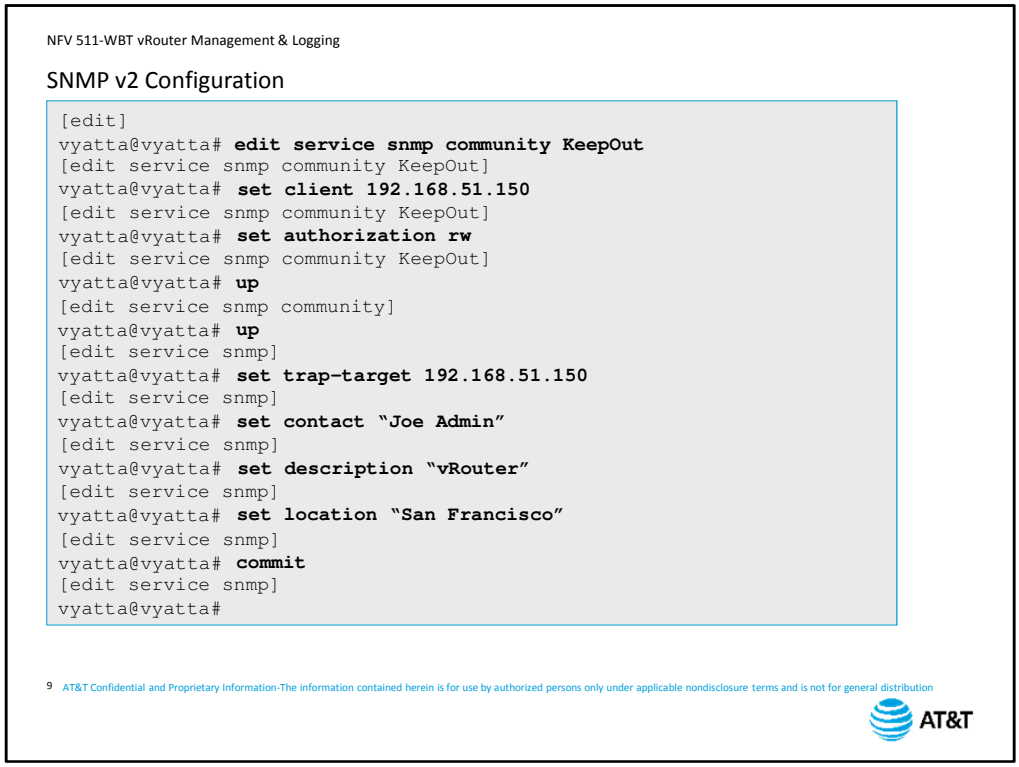
To enable SNMP on a vRouter, you need to define the SNMP community string. This is a string of characters that identifies a level of access to your vRouter. Your SNMP management station must have the same community string defined in order to communicate with the vRouter.
Because there are multiple parameters associated with the community string, we recommend using the edit command. This saves you some typing, and ensures all parameters are associated with the same string.
In edit mode, set the address of the SNMP management station associated with the community string. You can specify multiple client addresses.
Next, specify the privilege level for the community string – read/write, or read-only.
You can optionally set the address you want the vRouter to listen on for SNMP requests. If you do not configure this, the device will automatically listen on all active interfaces.



If you plan to have the vRouter send traps, you also need to specify the IP address of the SNMP manager receiving those traps.
Optionally, you can specify the source address for the traps. If you do not specify, the vRouter will use the IP address of the outbound interface sending the trap message.
You can set basic device information. When the SNMP management station first establishes communication with the vRouter, it will pull this information.



Let's look at a configuration example. In this scenario, our SNMP management station is at 192.168.51.150. We want messages to that station to originate from the address assigned to data plane interface 1 – 192.168.42.1.
The community string is KeepOut, with read-write permission.
All traps should be sent to the management station.
For the device information, the contact is *JoeAdmin*. The description is *Vyatta*, and the location is *San Francisco*.



We begin with the `edit` command shown. This command creates the community string *KeepOut*, and moves us within the configuration hierarchy to set parameters associated with the string.

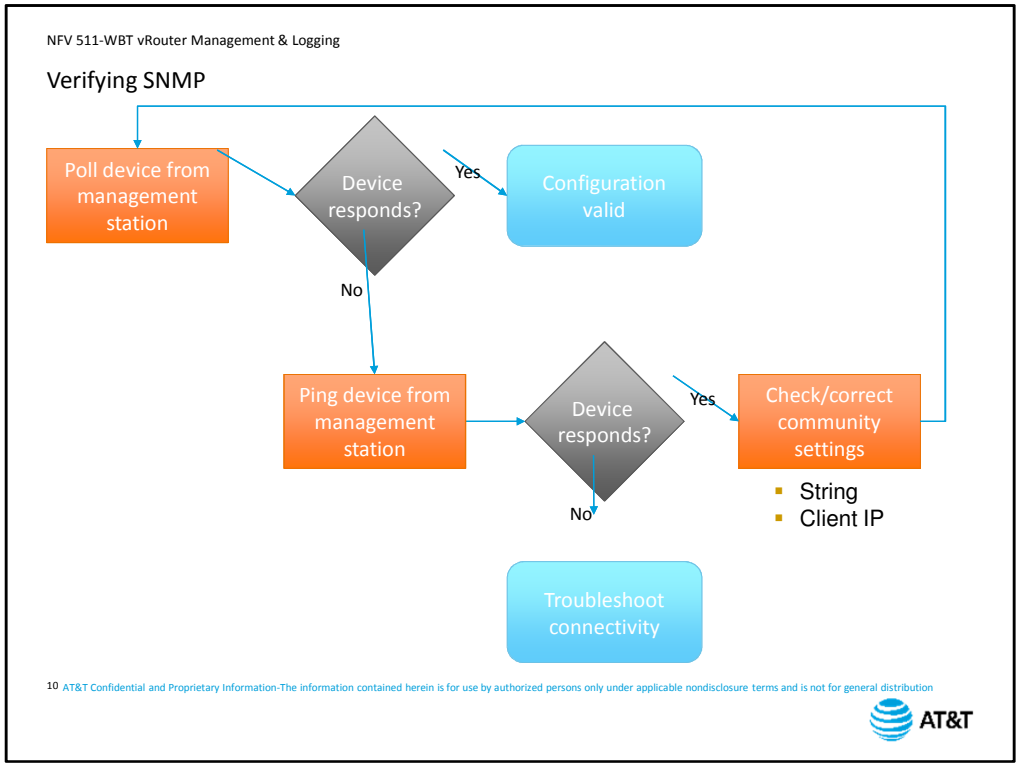
Next, we configure the address of the SNMP management station.

Next, we set the authorization to *read-write (rw)*.

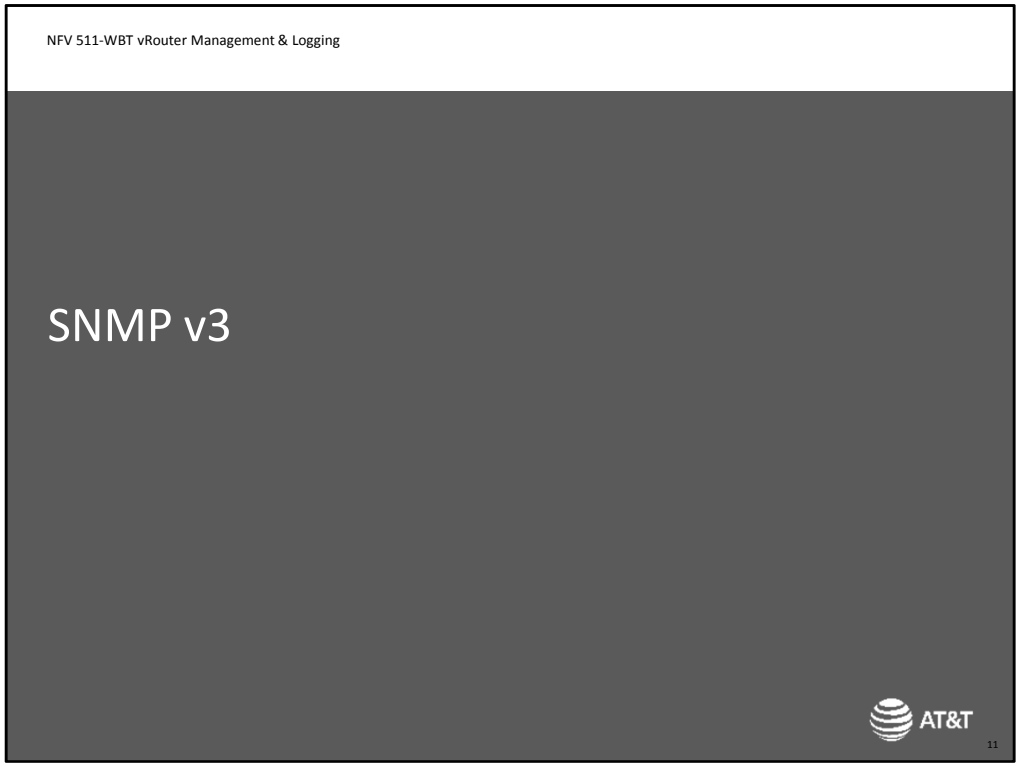
We go up two levels to exit the community string, then set the trap target. We do not need to set the trap source, since data plane interface 1 is the interface that will be generating the trap messages.

Finally, we set the contact information, the description, and the location. Note that we used quotes around the text so that the space characters are preserved.

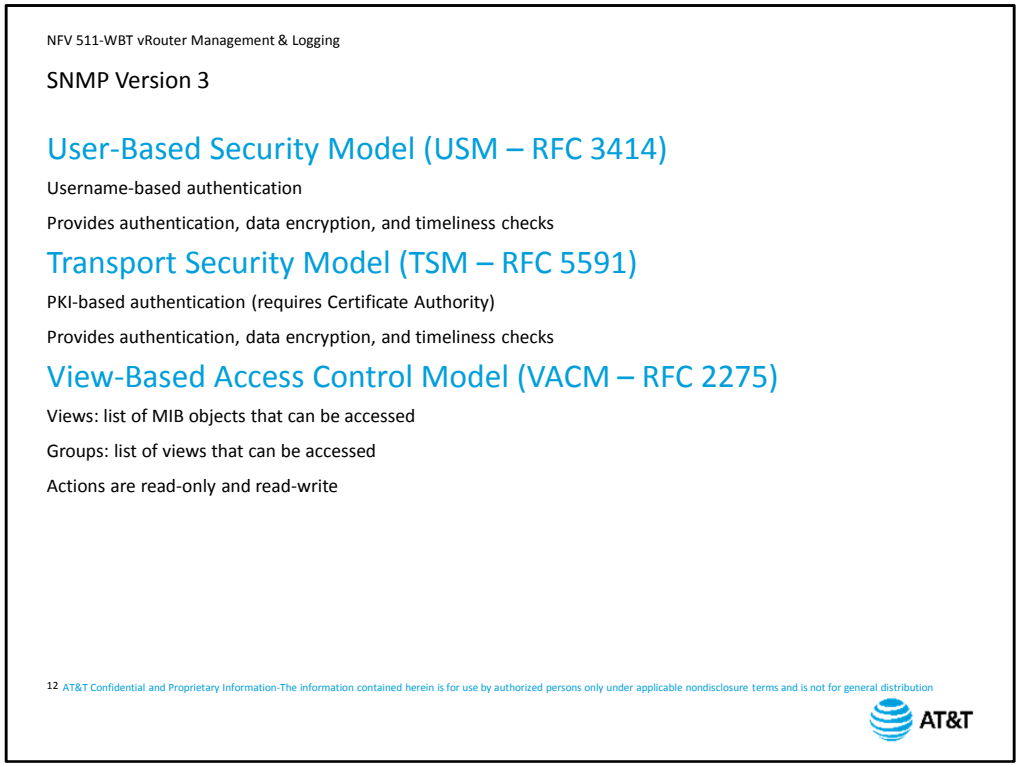
Finally, we commit our changes.



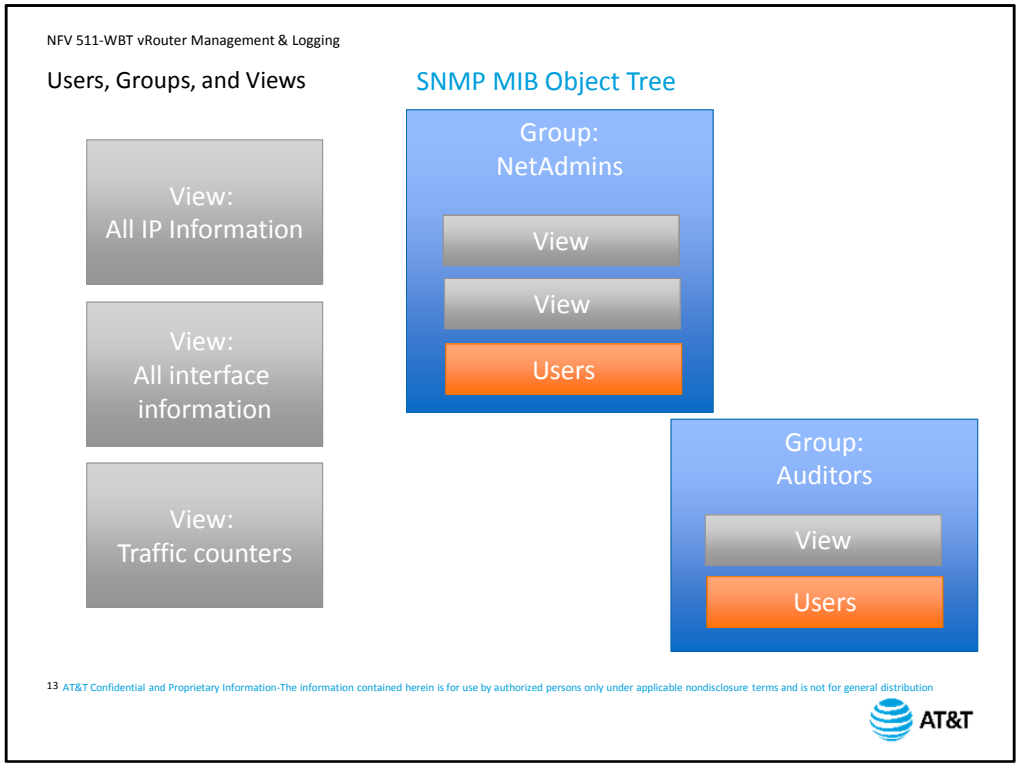
The best way to verify overall SNMP functionality is to poll your vRouter from your SNMP management station.
If the device responds to the poll, then your SNMP configuration is valid.
If it does not respond, then verify that basic IP connectivity exists by pinging the device from the management station.
If the device responds to the ping, then you need to verify the community settings on the vRouter.
Check the string for case sensitivity, and verify that the management station is configured as a client.
Retry the poll once you have corrected any misconfiguration. If the device does not respond to the ping, then you have an underlying IP connectivity issue. It could be a routing problem, or there could be a firewall blocking SNMP communications.



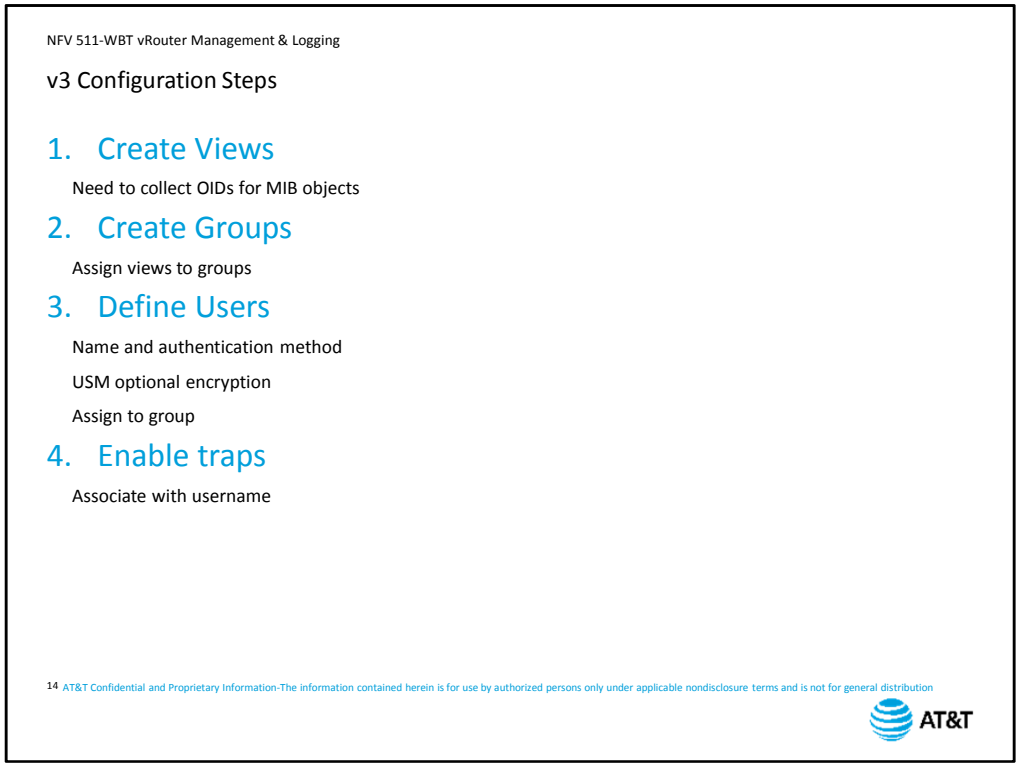
Now we will look at vRouter support for SNMP version 3.



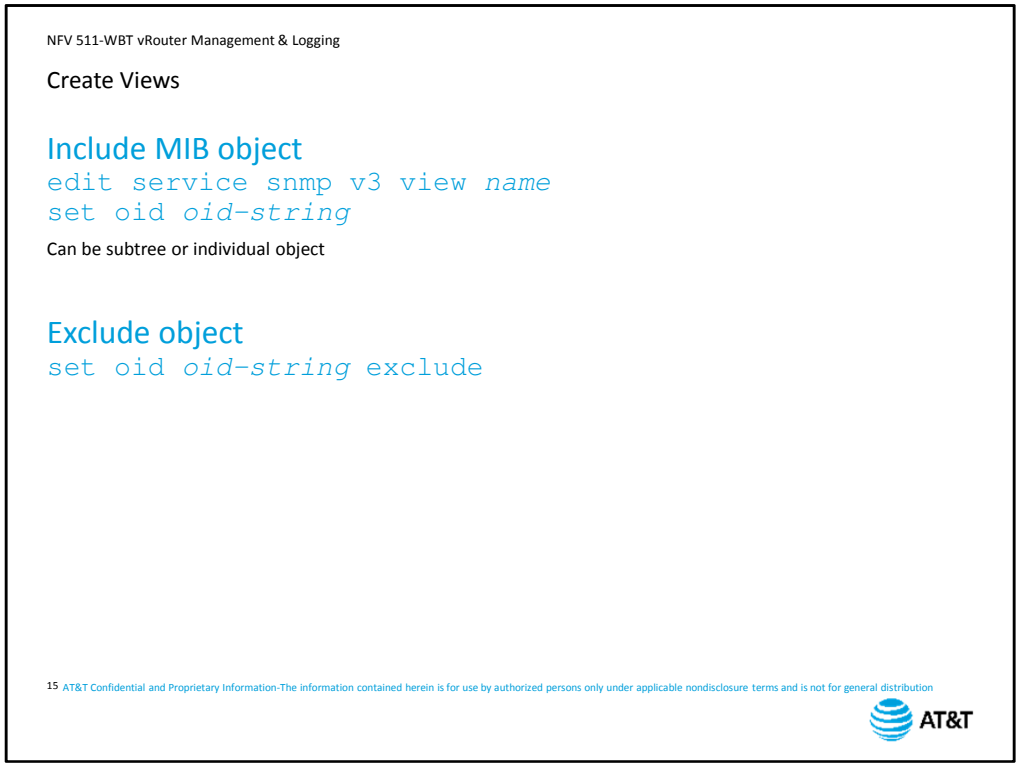
SNMP version 3 made significant improvements in security and manageability by introducing three major concepts.
The first is the User-Based Security Model or USM.
Instead of having generic “group” keys as with version 2, version 3 allows you to define individual username/password combinations for user authentication.
You can optionally enable encryption. The standard also defines timeliness checks. Since SNMP messages are often time-critical, making sure that messages are current is important for accuracy of information.
The second is the Transport Security Model or TSM which uses the Public Key Infrastructure for access authentication and encryption. PKI uses certificates to identify devices, and those certificates must be generated and managed by a Certificate Authority.
TSM provides the same benefits as USM, using certificates rather than username/password strings.
Finally, version 3 introduced the concept of View-Based Access Control
You, the administrator, can group specific MIB objects into views, then use Groups to define who can access a view or set of views.
For each group, you can specify an action of read-write or read-only.



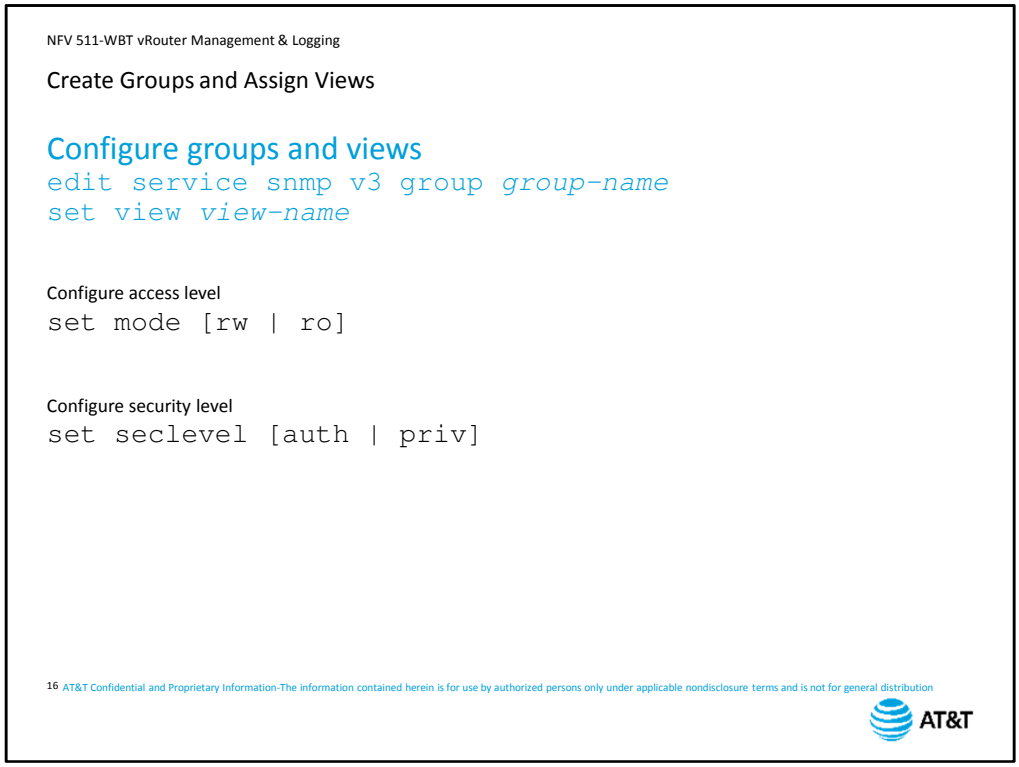
Let's look at how users, groups, and views fit together.
A view is a list of SNMP Object Identifiers, or OIDs. You, the administrator, determine how you want to group these OIDs.
The OID structure is a tree, so if you include an object with sub-branches, those sub-branches will also be included in the view.
A group is a list of views. You can have one or more views per group, and you can use the same view in multiple groups.
You then assign individual users to groups. A user can only be assigned to a single group.



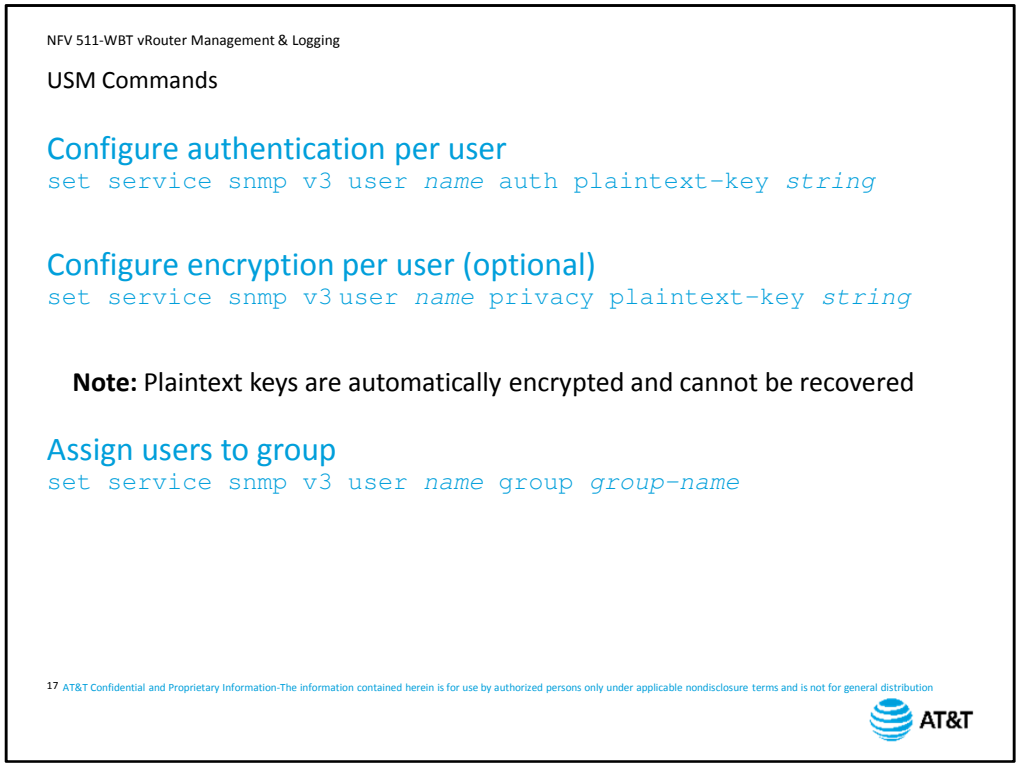
To configure a vRouter for operations with SNMP version 3, follow these steps.
First, you need to create your views. You need to have a MIB browser or a list of OIDs available in order to configure the views.
Next, create the groups and assign the views you want available to each group.
Next, define the SNMP users. You can use both the User-based Security Model and the Transport Security Model. You'll need to define the authentication method, optionally enable encryption, and assign the user to a group.
Finally, you can enable SNMP traps. Traps are also associated with usernames.



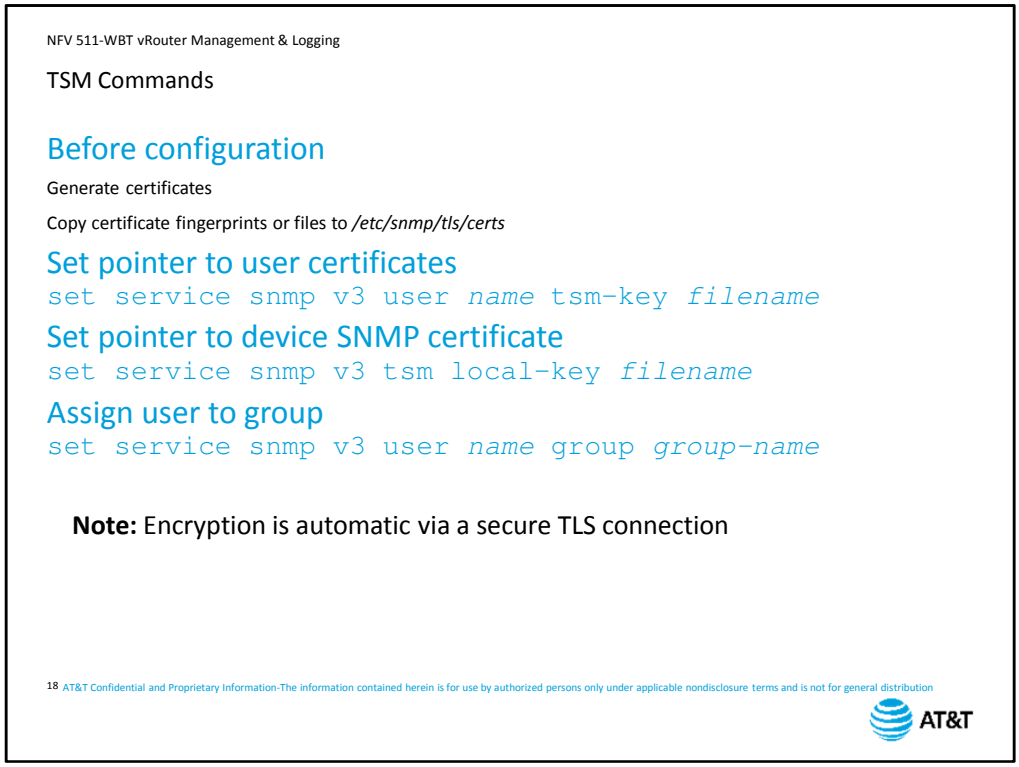
To create a view, use the command `set service snmp v3 view`. Specify the name of the view and the OID you want included in the view.
If you plan to include several OIDs in a view, it is easier to use the `edit` command to create the view and move within the configuration hierarchy, then use the set command to just specify the OIDs.
Remember, the OID can represent an individual MIB object, or can represent a sub-tree of objects.
If you include a tree, but do not want some of the objects within that tree, you can use the `exclude` parameter for those objects you do not want included.



The next step is to create the group and assign the views. If you have only one view for a group, you can do this in a single command.
If you have multiple views, again it may be easier to use the `edit` command to create the group, then set each view within the group.
Set the group for read-write or read-only access.
You also need to specify the security requirements for the group. All users assigned to the group must have the same security level as you set here.

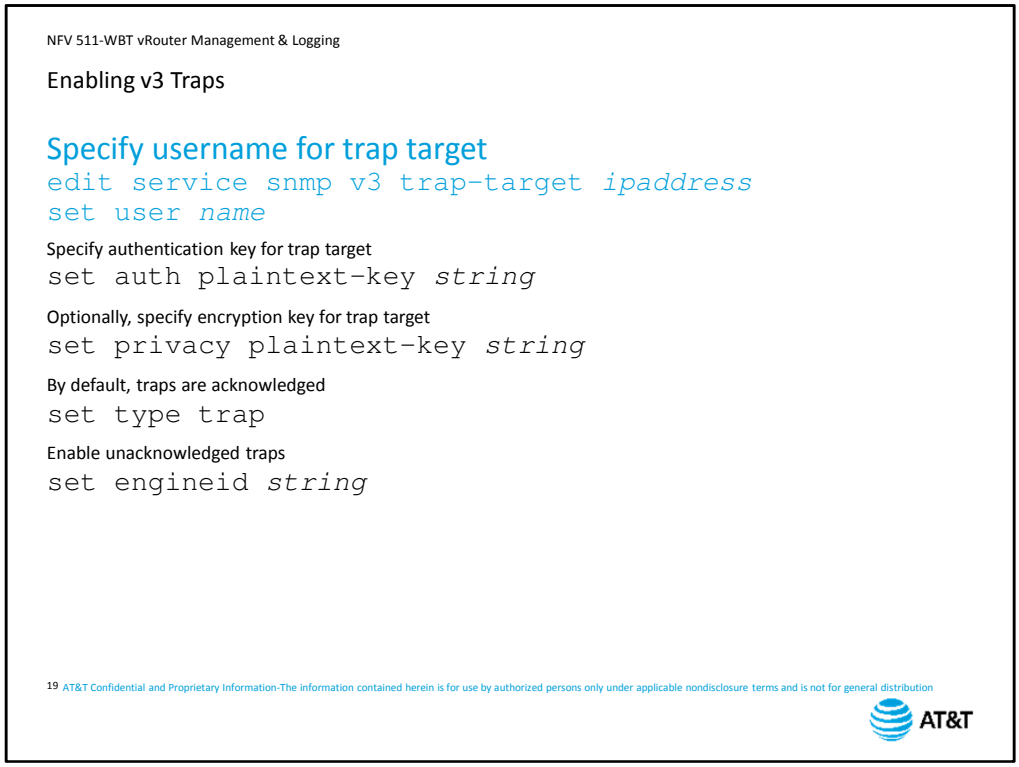


The next step is to define our users. If using User-based Security Model or USM, you create the username and set the password in a single command.
Optionally, you can configure an encryption key. Both the password and the encryption key must be identical on both the vRouter and the SNMP management station.
Although you configure these keys in plain text, they are encrypted automatically by the vRouter and stored in the configuration file in an encrypted format. You would have to decrypt the password to recover it.
Next assign the user to a group, using the group name you defined in the previous configuration step.

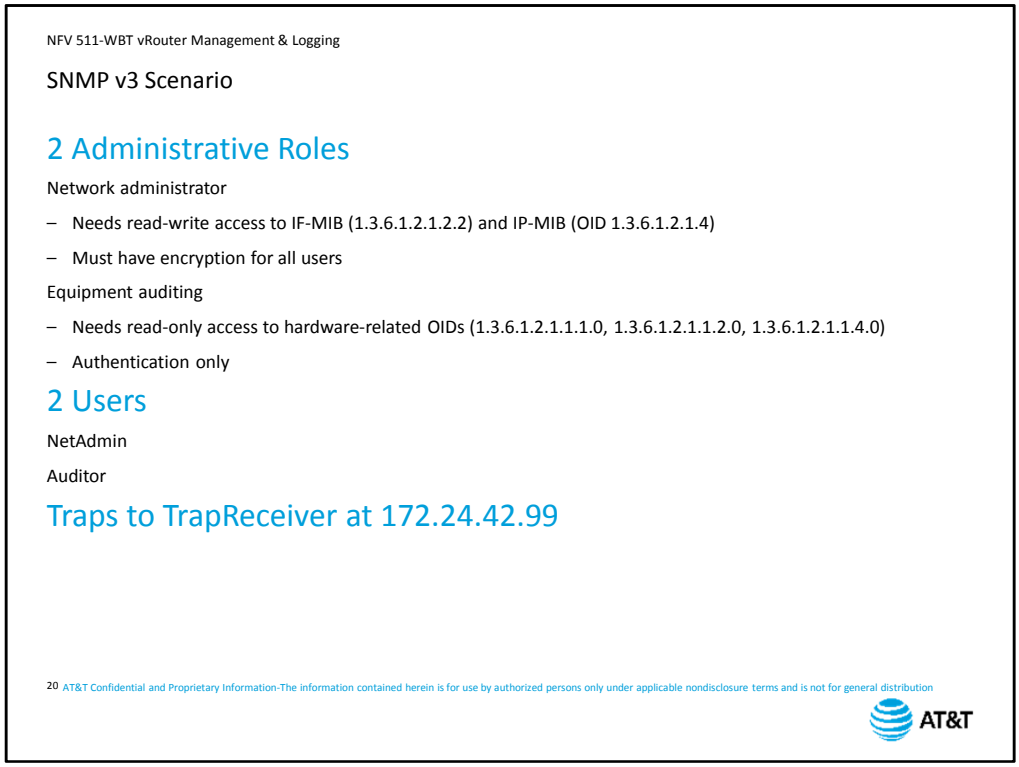


If using or TSM, you need to generate certificates for all users and for the vRouter. Copy the certificates to the */etc/snmp/tls/certs* directory on the vRouter.
Within the configuration file, set the pointer to the appropriate certificate file for each username.
Next, set the pointer to the certificate for the vRouter itself.
Finally, assign the TLS user to the appropriate group.

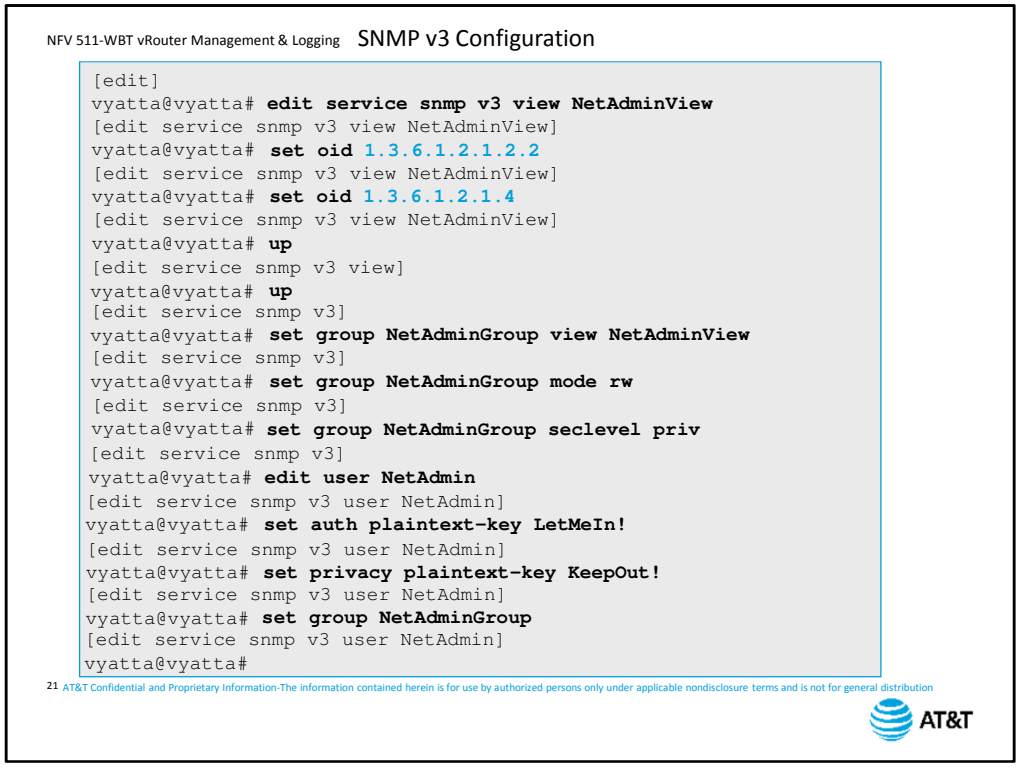
You do not need to set any encryption keys; the PKI exchange establishes a secured TLS connection between the vRouter and the SNMP management station automatically.



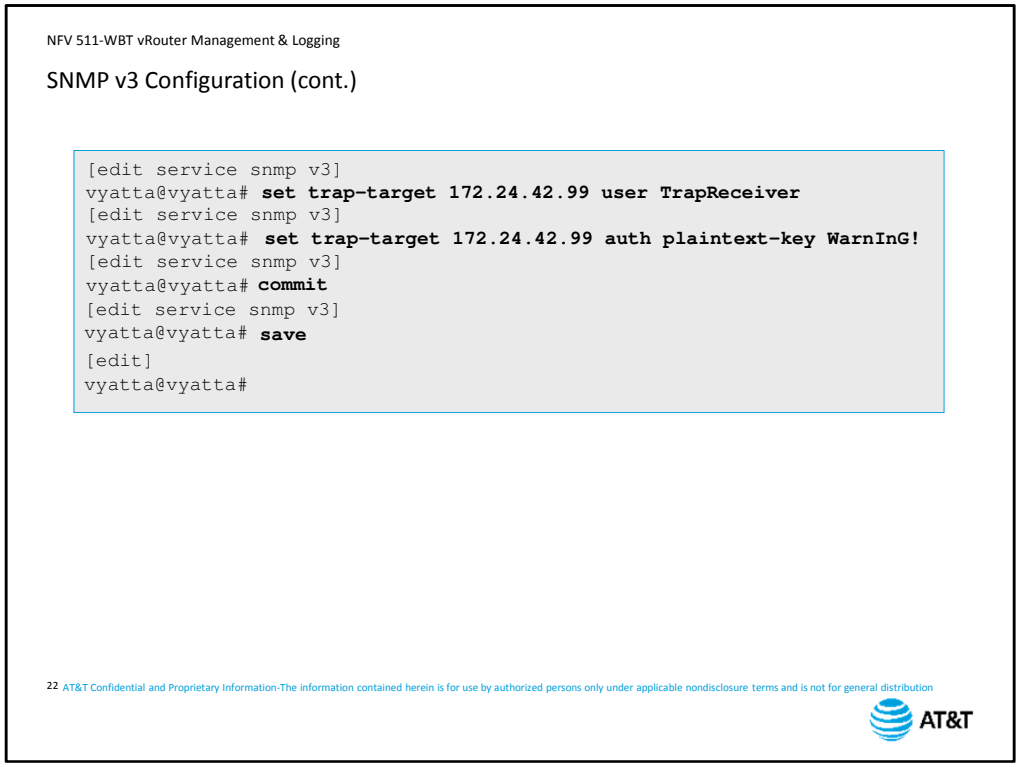
Version 3 traps need to be associated with a specific username. This does not have to be a user you've already defined on the vRouter, but it does have to be defined on the SNMP management station. It is a best practice to have a separate user identity for traps.
You also need to specify the authentication key for traps.
If you want traps sent in encrypted format, set the encryption key. As with other keys, both authentication and encryption are configured in plain text, but stored in encrypted format. Keys must be identical on the vRouter and the SNMP management station.
By default, traps are sent using UDP and are unacknowledged. If you want traps to be acknowledged, you need to configure `type trap`, then set the Engine ID for the device generating the trap responses.



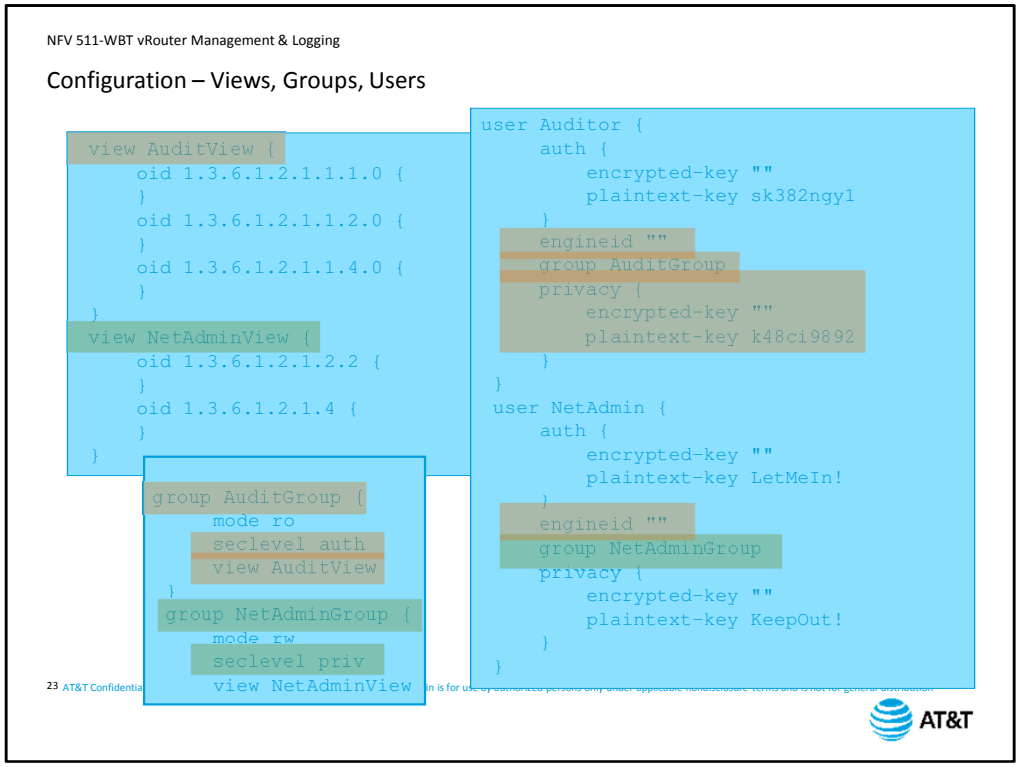
In our sample scenario, we have two separate administrative roles.
The Network Administrator needs read-write access to the complete IF-MIB and the IP-MIB. We have already looked up the OIDs.
Users in this group must have encryption as well as authentication.
The Equipment Auditor role needs read-only access to specific hardware-related OIDs.
Users in this group only require authentication.
We have two users on our SNMP management station, Netadmin and Auditor, that we will assign to our groups.
Traps will go to the TrapReceiver user on the management station at 172.24.42.99.



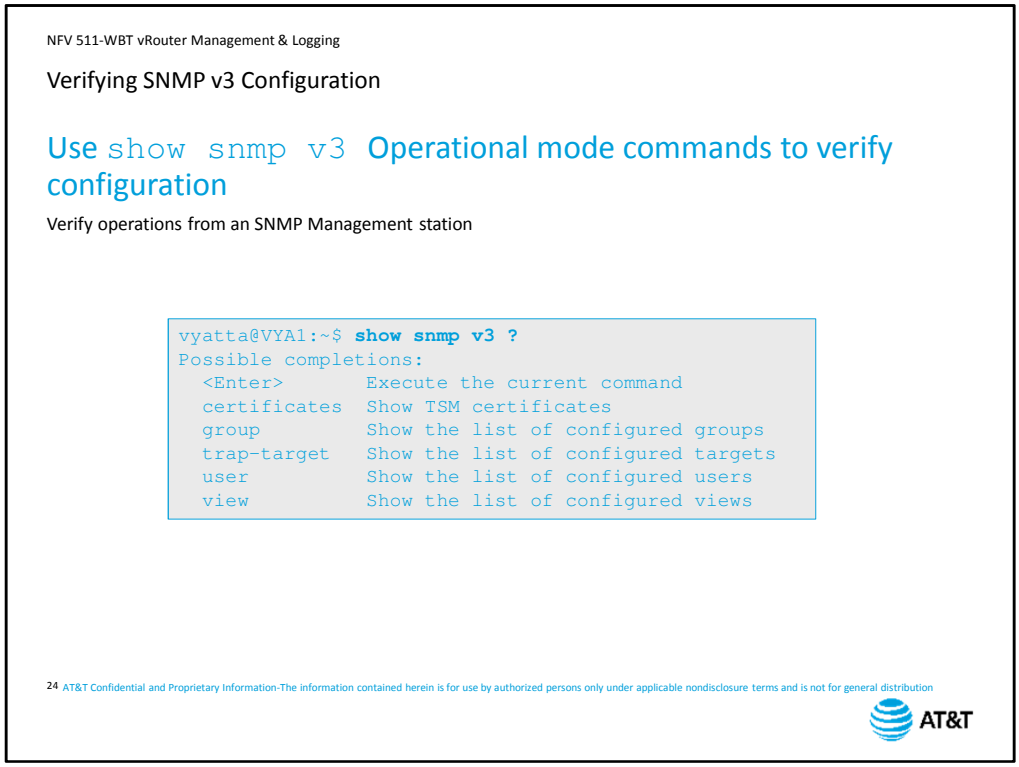
We configure the first set of view, groups, and users so you can see the syntax in action, then look at the complete configuration.
We begin with creating the NetAdmin view, then assigning the two OIDs to the view.
We go up two levels to exit the view level of the hierarchy.
Next, we create the group and assign the view in a single command.
Then configure the mode of the group to read-write, and set the security level to use encryption. We need to make sure that all users assigned to this group also have encryption configured.
Because we are setting a few parameters for the user, we use the `edit` command to create the username, set the authentication key,
Then set the encryption key. Note that the authentication and encryption keys must be at least 8 characters.
Finally, we assign the user to a group. We repeat the same steps for the second view, group, and user.



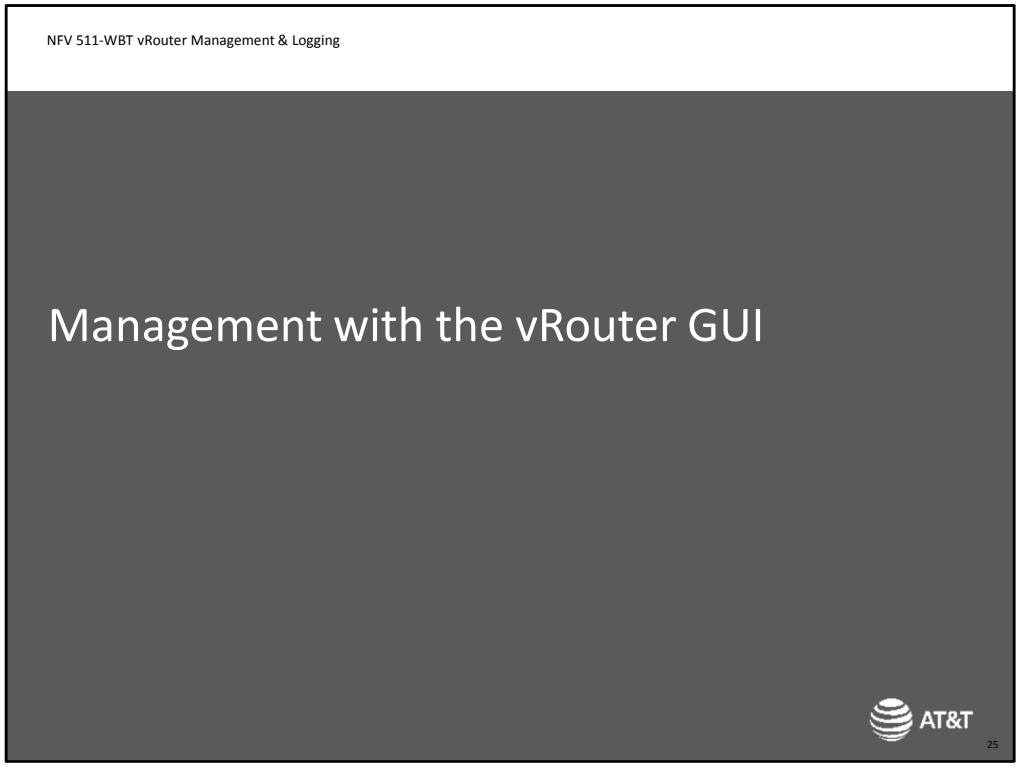
To finish our configuration, we start at the `service snmp v3` level of the hierarchy and set the trap target address and username.
Next assign the authentication key. Remember the key must be at least 8 characters. Note, we will not be encrypting or acknowledging trap messages.
We commit our configuration, then save it.



When we look at the both views, we see the list of OIDs assigned to each view.
The views are then associated with the groups we created, along with the mode and security level
The users are associated with the groups. Because we have not yet established a connection with an SNMP management station, the key strings are still in plaintext.
The engine ID field will be populated once a connection is established, and identifies the specific management station that authenticated the user.
Also note that, although the Audit group security level is set to authentication, the Audit user still has a privacy key configured. This works because the user is more secure than the group definition requires. If you try to assign a user with authentication-only security to a group that requires encryption, you will get a commit error.

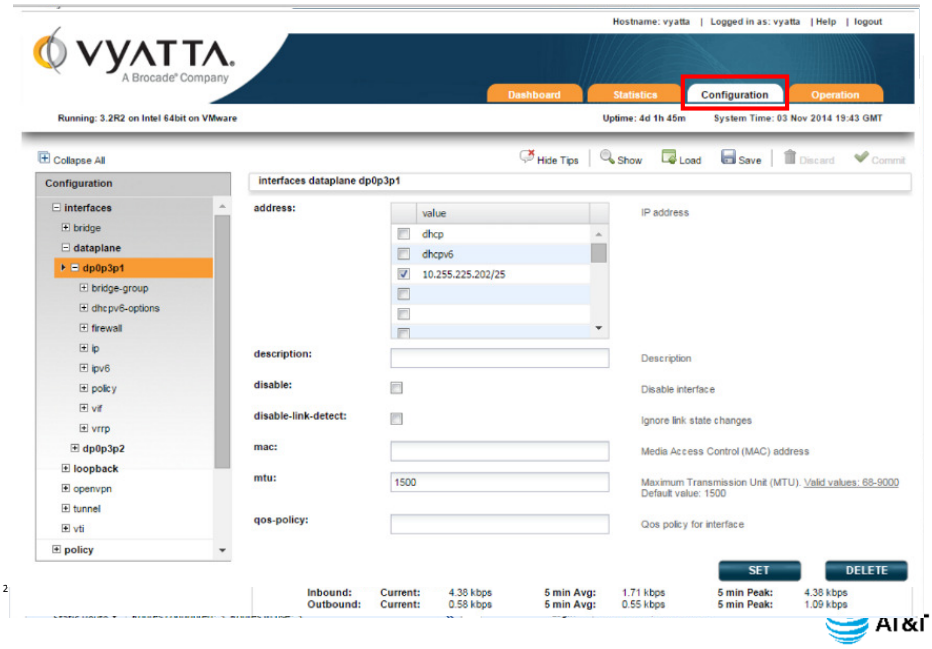


As with SNMP version 2, the vRouter can only verify your configuration data from the CLI. You can use show commands to display the list of known certificates, group configuration, trap target configuration, users, and view configuration.
To verify actual operations, you will need to use your management station to test the access to the vRouter.

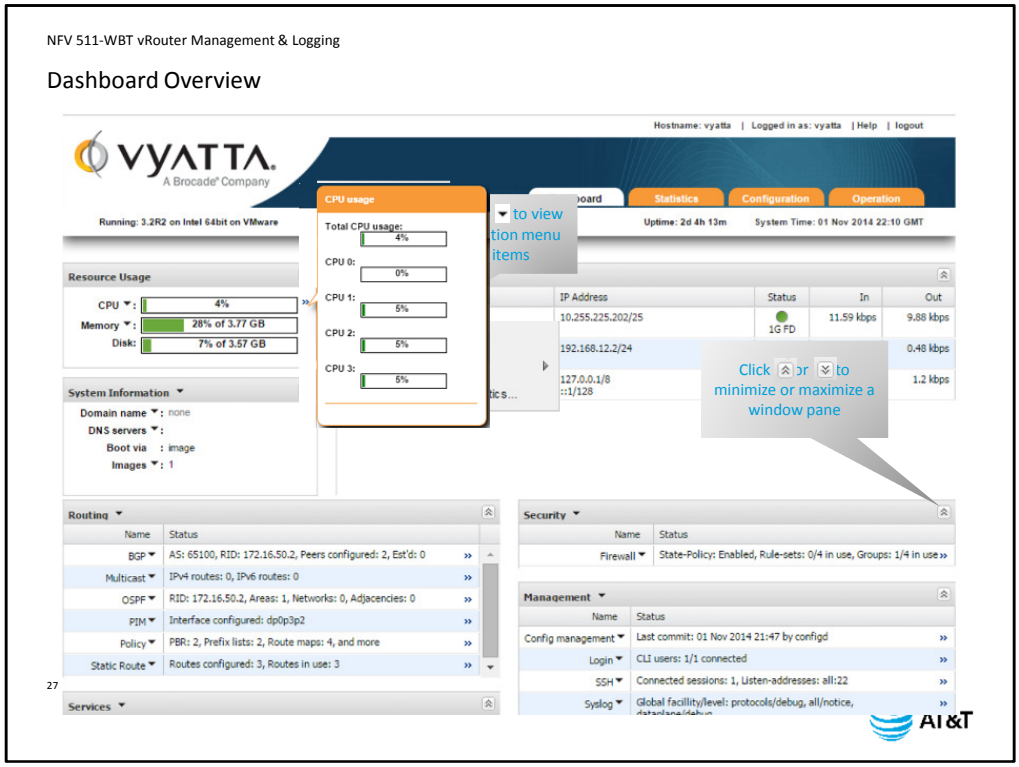


NFV 511-WBT vRouter Management & Logging

The vRouter GUI



The vRouter GUI provides you with an easy-to-use graphical view of your vRouter.
The Dashboard tab gives you a single-screen overview of what is happening on your vRouter. We will go through the Dashboard windows in more detail on the next slide.
The Statistics tab allows you to view interface and resource utilization graphs. The graph updates every 5 seconds, and displays a sliding 5 minute window.
The Configuration tab provides you with a drop-down menu on the left to navigate the configuration hierarchy, then fields to enter configuration parameters.
The Operation mode tab provides you with a drop-down menu on the left to navigate Operational mode commands, then displays the output on the right.
Because this course is about managing your 5600 vRouter, we are going to focus on the Dashboard tab.



The vRouter GUI Dashboard provides you with one-stop monitoring for your vRouter. Information about key features and functionality are summarized on a single screen, making it easy to see the enabled features and overall device function in one place. Another feature of the Dashboard is the ability to view additional information about a particular object without having to open a new window.

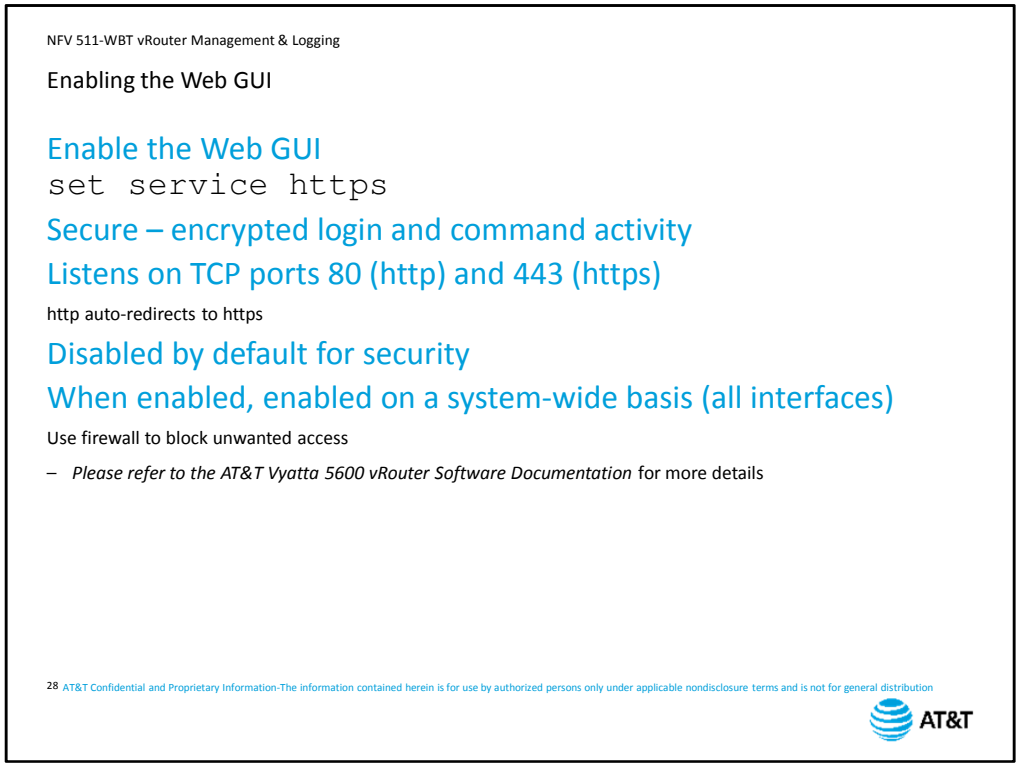
Blue double-right arrows next to a displayed item indicates that more details are available. Click on the arrows, and a pop-up window appears with the additional information – in this case, the summary of the CPU usage. Clicking anywhere on the main screen causes the pop-up to disappear

A black down-arrow indicates that there are jump-to options available. The actual options depend on the feature or function.

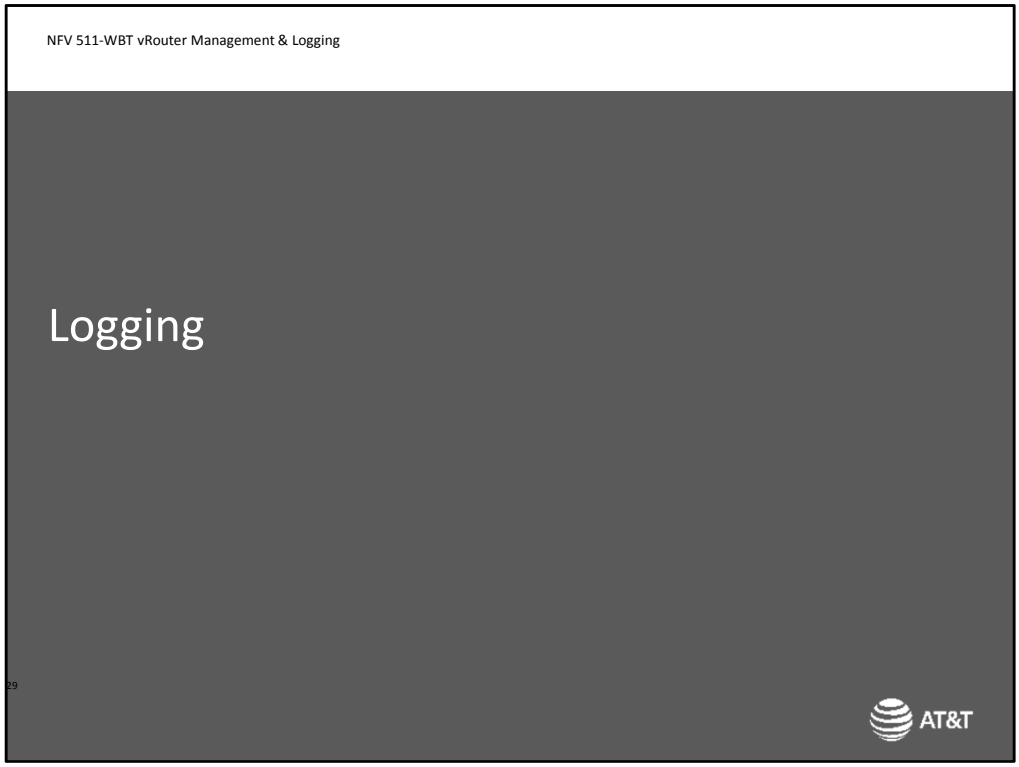
Edit will jump you to the configuration tab, with the configuration menu hierarchy set to the appropriate level for the feature or device. In this case, clicking on *Edit* would take you to the “set interfaces data plane 1” level.

Run will jump you to the Operation tab, with the selected command – in this case, either clear or show – selected and ready to run. *Statistics* will jump you to the Statistics tab and display the appropriate graph.

You can customize your Dashboard view by minimizing or maximizing the specific window panes. For example, click the double up arrows next to *Security* and the pane will hide from your view.

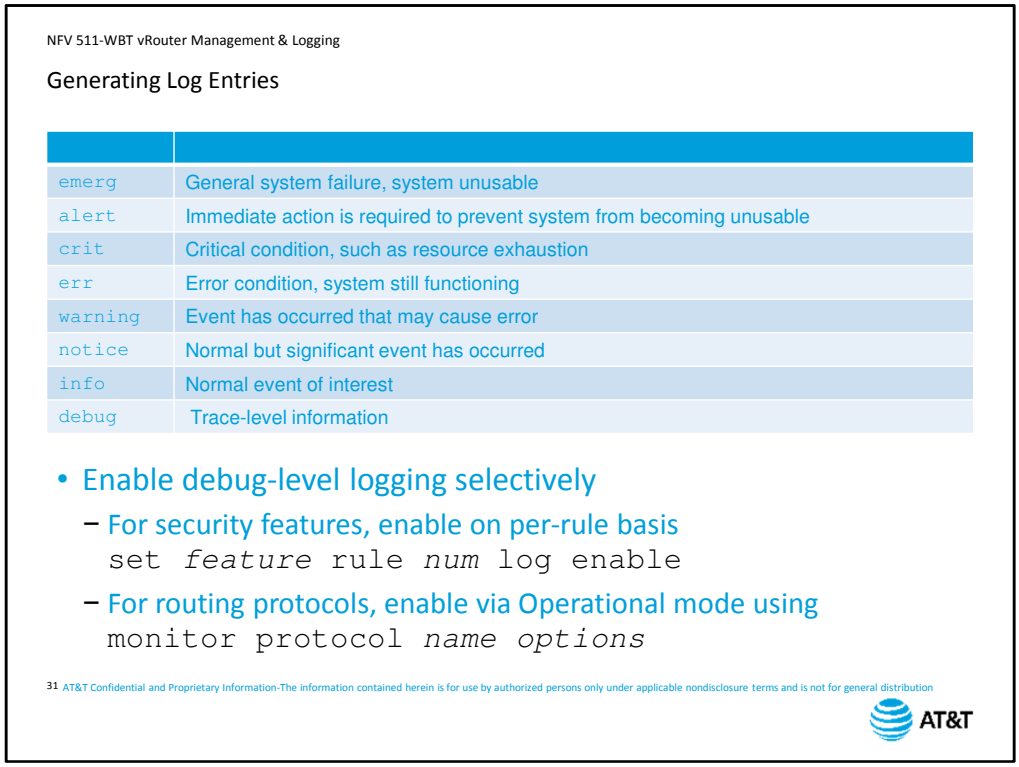


To enable the Web GUI, use the command `set service https`.
The GUI provides encrypted transport of both login and command activity.
The Web GUI listens on TCP port 80 (for HTTP) and port 443 (for HTTPS).
Any HTTP open request is automatically redirected to the HTTPS port to open a secured connection.
As with other network-based access, the Web GUI is disabled by default for security.
When you enable it, it is enabled system-wide basis. That is, it is available through all network interfaces.
You need to configure the vRouter firewall feature to block unwanted access. For more details on firewall configuration, please refer to the *AT&T Vyatta 5600 vRouter Software Documentation*.

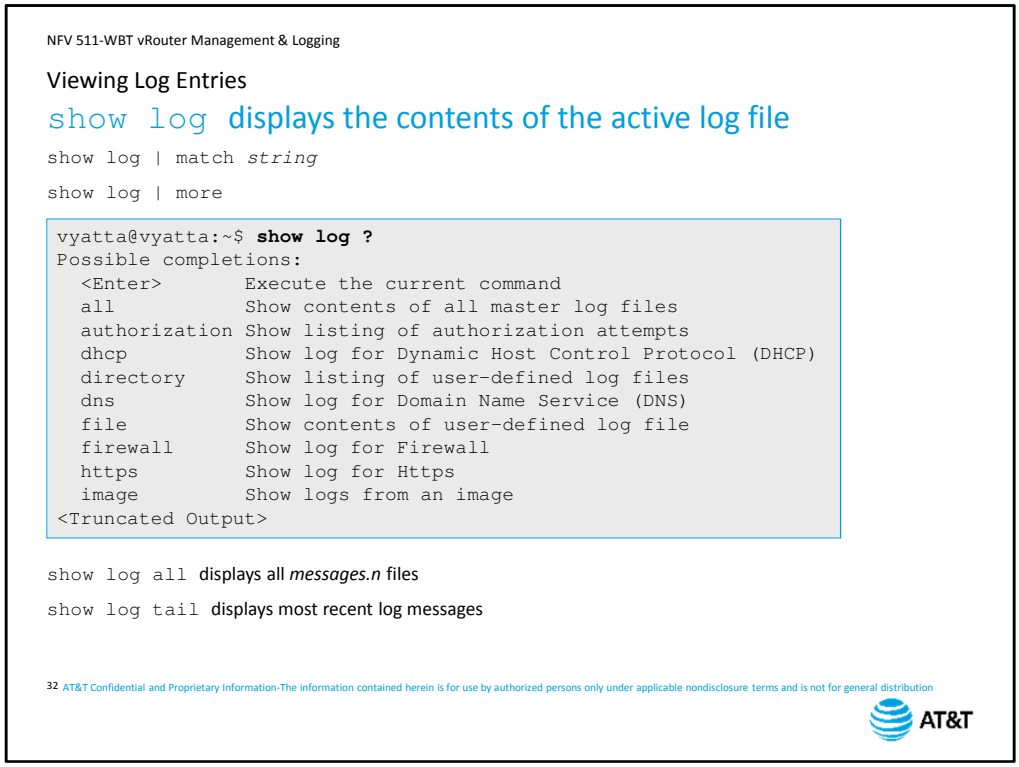


Next we'll look at the logging capabilities of the 5600 vRouter.

One of the advantages of vRouter software is its extensive local logging capabilities. Unlike traditional networking devices, vRouter supports an integrated hard drive, providing capacity for storing local messages without the requirement of an external server.
vRouter software uses the Linux standard syslogd process to store logs.
Log messages are stored in the file */var/log/messages*.
When this file reaches 500KB in size, the vRouter renames the file to *messages.0* and opens a new messages file. This continues up to the file *messages.9*, for a total of 5 gigabytes of log messages.
In addition, the vRouter system maintains separate logs for bootup messages, PPP connection setup, IPsec connection setup, and other features.

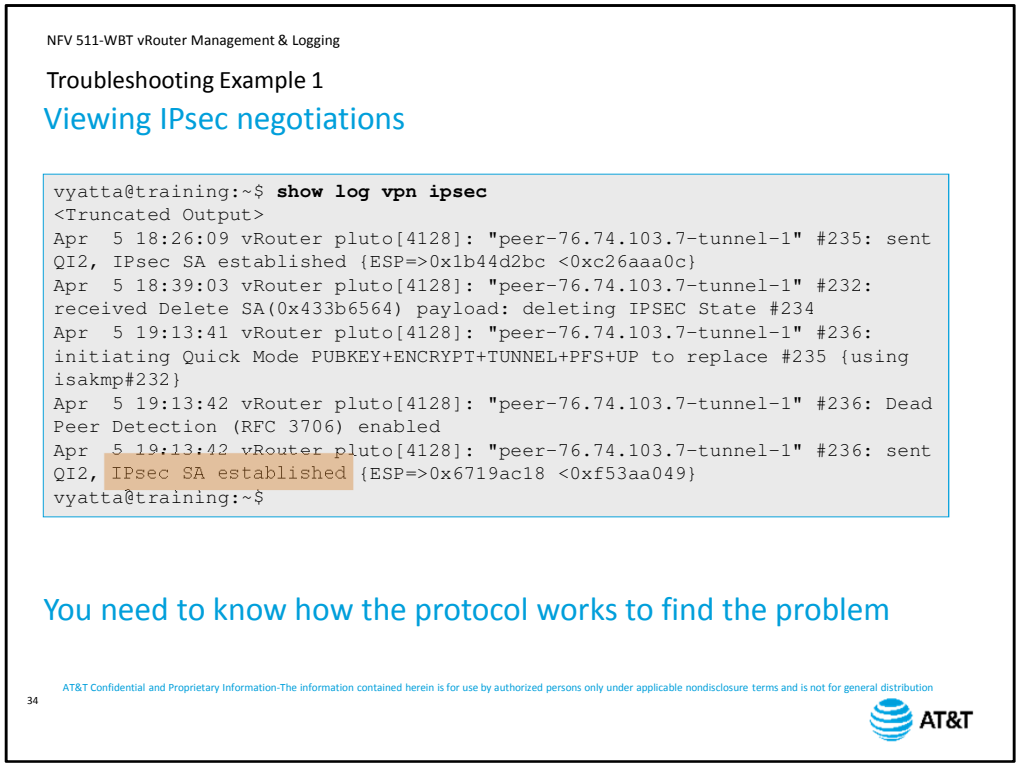


Syslog has 8 pre-defined levels for log messages.
The default logging level is notice, which means that by default, the system log records messages with a level of notice or higher.
In some cases, particularly if you are troubleshooting, you may want to capture trace-level information – that is, packet-level captures of protocol exchanges or security features. To do this, you enable debug-level logging selectively on the feature you want to examine more closely.
For security features such as NAT and firewall, you can enable debug-level logging on a per-rule basis.
For routing protocols, you enable debug-level logging from Operational mode using the `monitor` command. We will examine examples of both.

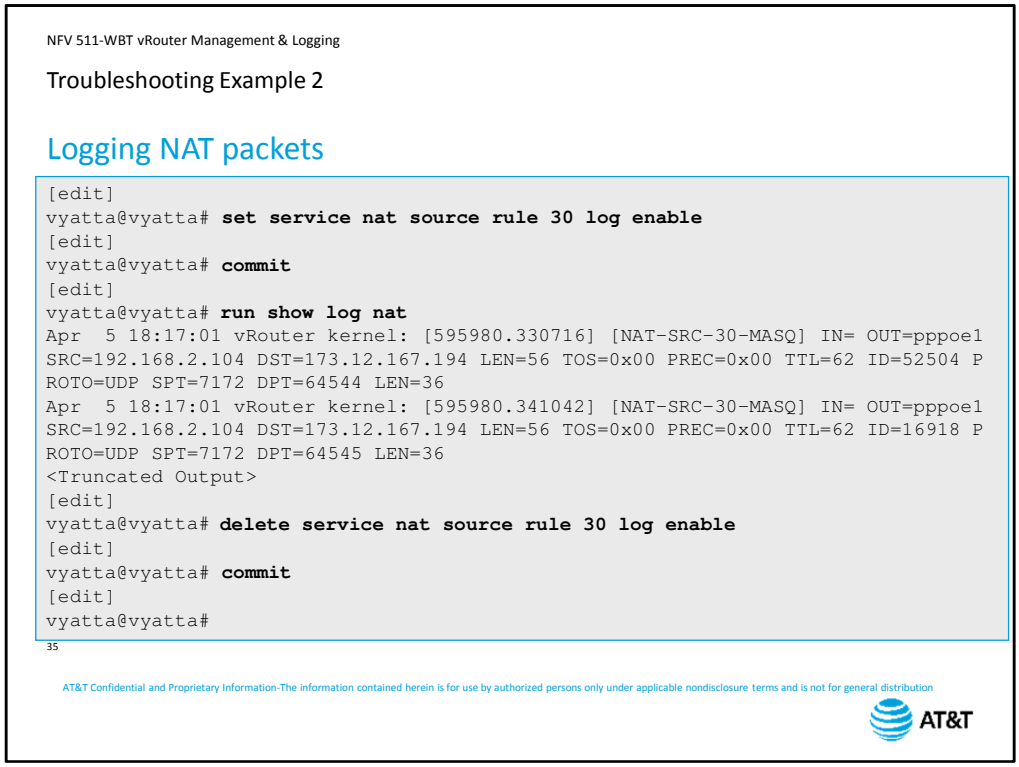


To view the active log file, use the Operational mode command `show log`.
Because this can be up to 500kilobytes of messages, we have pre-defined some filters for you. By adding the appropriate option, you will only see messages relating to the specified feature or protocol.
You can also use the Linux-style pipe (|) and specify a specific text string to search for. The device will display all lines in the log that contain the string.
If the output is more than one screen, use pipe (|) and `more` to display one screen of data at a time.
To view the entire set of log files in sequential order, use the command `show log all`. Again, you can use the match tool to search the output.
If you just want to see what happened most recently on your device, you can use the `show log tail` command. This displays the last 10 log entries.

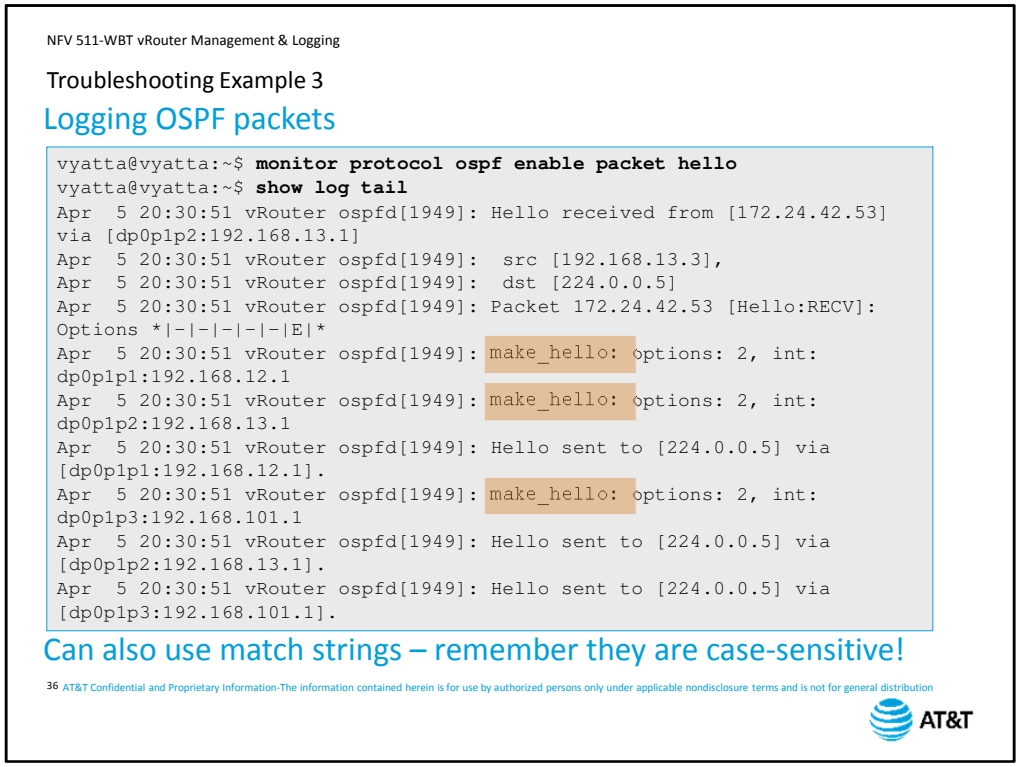
Here is an example of using Linux pipes (|), with the `match` and the `more` commands. In this case, we are searching for all log messages that contain the string *ERROR*, and limiting the display to one screen at a time.



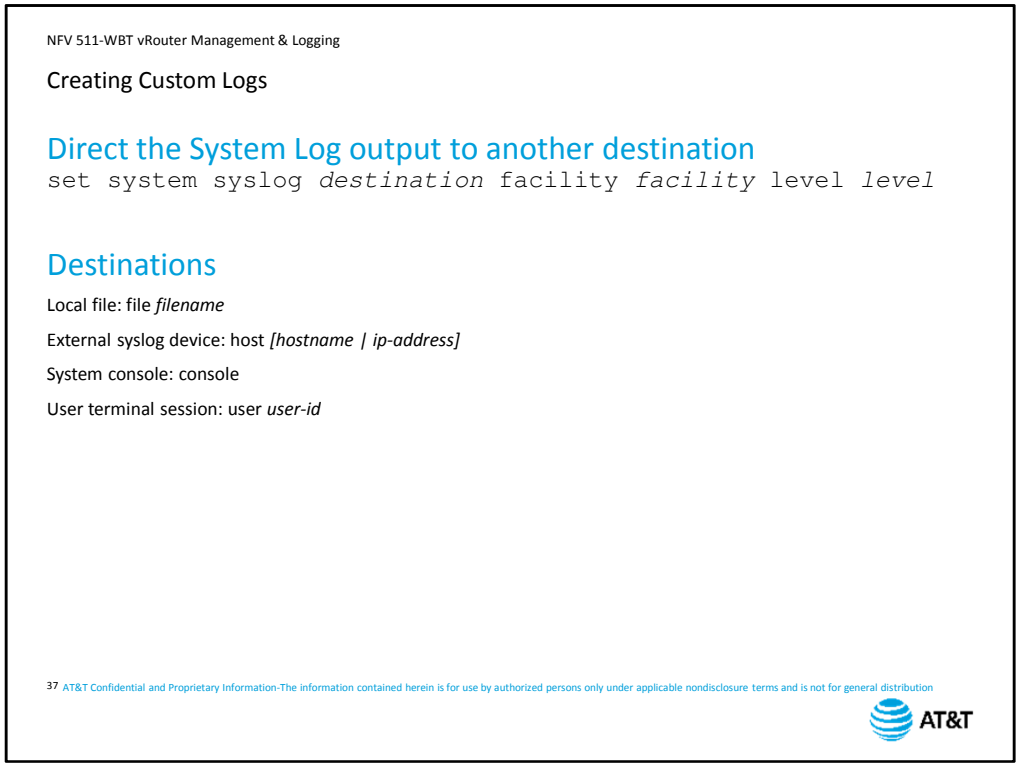
Here is an example of the VPN logs. The output displays the last few messages of a successful IPsec tunnel negotiation.
If the text on the screen does not make any sense to you, then it is illustrating an important point of using log output. You need to know how the protocol you are trying to troubleshoot is supposed to work in order to understand the log messages.
In this case, if you do not know that IPsec negotiations end when an SA is established, you do not know that the last message indicates a successful tunnel negotiation.



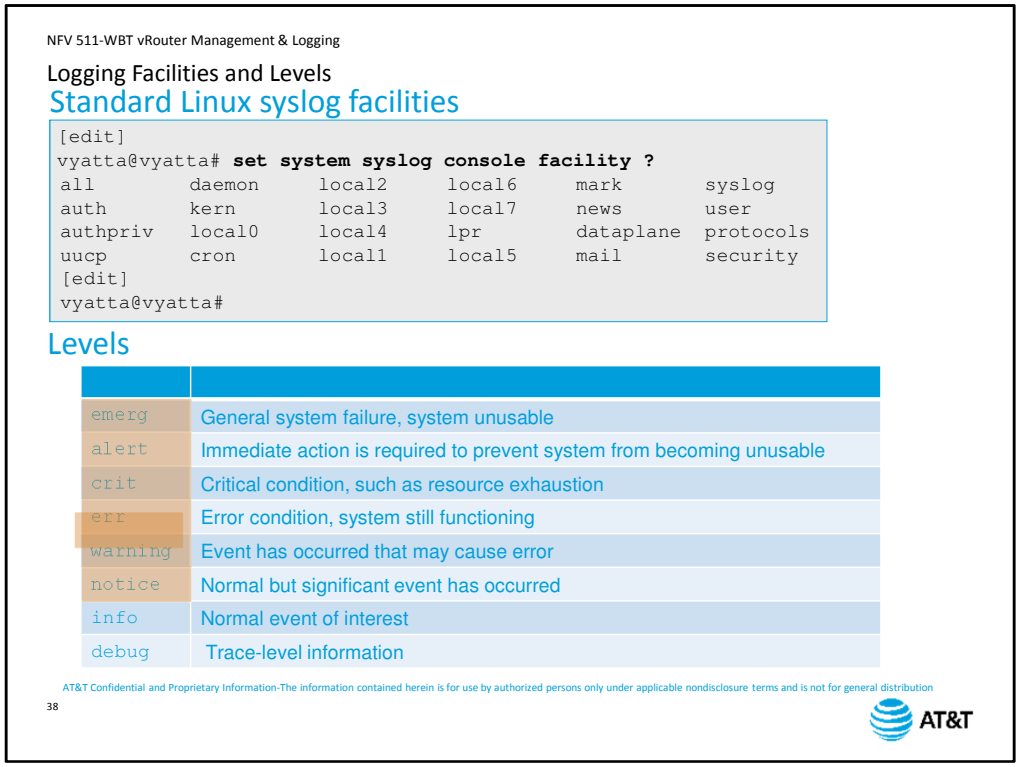
In this example, we are enabling logging of a specific NAT rule. We enable logging on `nat source rule 30`, then commit the change to make it active. Now we wait a bit for log entries to be generated as traffic passes through the vRouter and matches the rule.
Now we can look at the log to see if any traffic has matched our logged rule. Since we are in Configuration mode, we have to add the run parameter to the `show log` command. This lets us see the log data without having to exit Configuration mode. In this case, logging captures the headers from the packets that match the rule.
When we are done, we have to remember to delete logging from the rule, or the vRouter will continue to log packet headers that match the rule.
We have to commit the deletion in order for it to take effect.



In this example, we enable logging for OSPF hello packets. Unlike the security features, routing protocol debug-level logging is enabled in Operational mode.
The `monitor protocol` options provide several choices for each routing protocol. We are interested in OSPF hello packets, just for the purposes of demonstration.
To view messages, we use `show log tail`. We know OSPF hellos arrive every 10 seconds, so at least some of the last logged messages should be OSPF hellos.
We could also use a pipe (|) and `match` command to search for a specific text string, but remember that those searches are case-sensitive.
If we searched for *hello*, all lower-case, only those lines with the highlighted match would be displayed in our output. We would miss all the other hello-related messages.



In addition to the default system log, you can direct the system's log output to other destinations using the `set system log` command.
Options for `destination` include:
`file` - a local file which you can then read using the `show log` command. This file is in addition to the default local log already created by the vRouter.
`host` - sends messages to an external server running the syslogd utility.
`Console` - directs log output to the system console.
`User` - directs log output to a the terminal of a specified user session.
You can also use `set system syslog global` to direct output to the system standard location.



You can limit the contents of your custom log by specifying a particular syslog facility to capture. Note that these are syslog facilities, not specific network protocols or vRouter features.
We mentioned the logging levels earlier. Setting a level includes all higher levels,
So if you set the log to *notice* messages, the log will capture messages from *emergency* through *notice*.

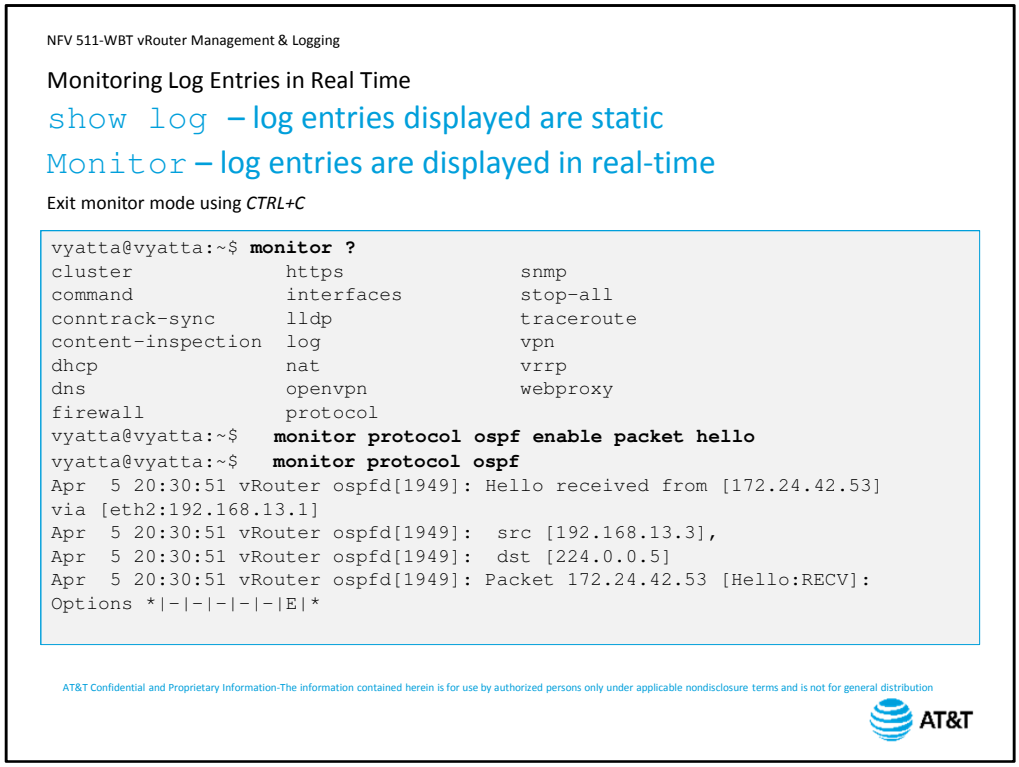
NFV 511-WBT vRouter Management & Logging

Custom Log Scenario

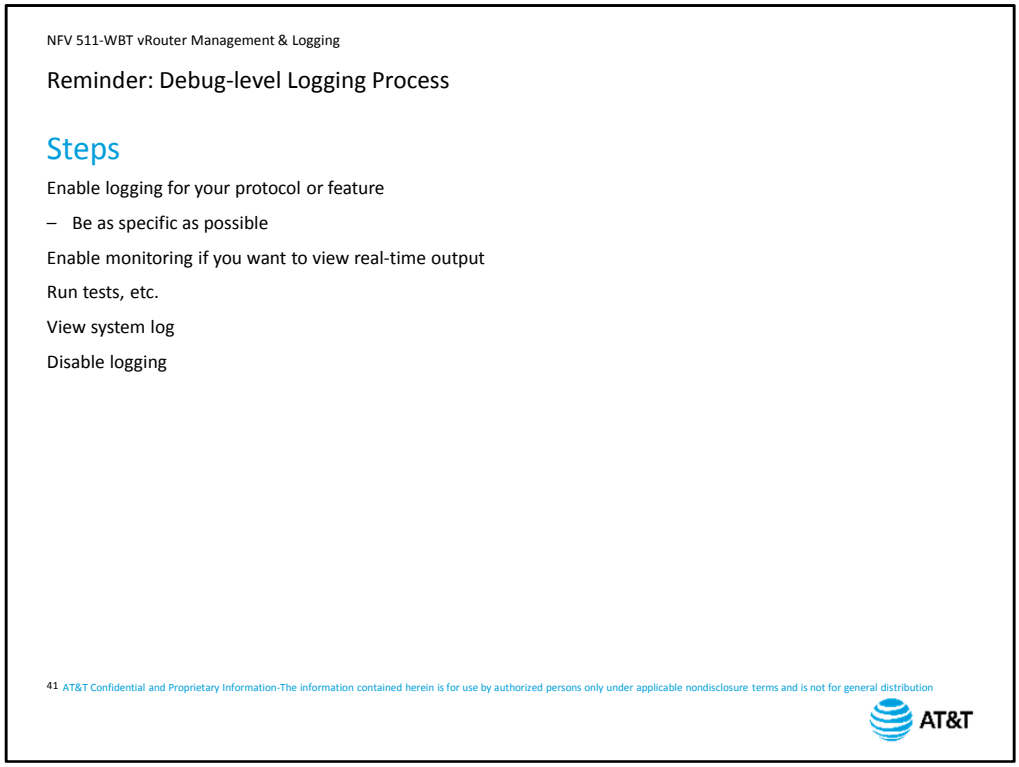
Send error messages to external syslog server, facility 0


```
[edit]
vyatta@vyatta# set system syslog host 192.168.51.150 facility local0 level err
[edit]
vyatta@vyatta#
```

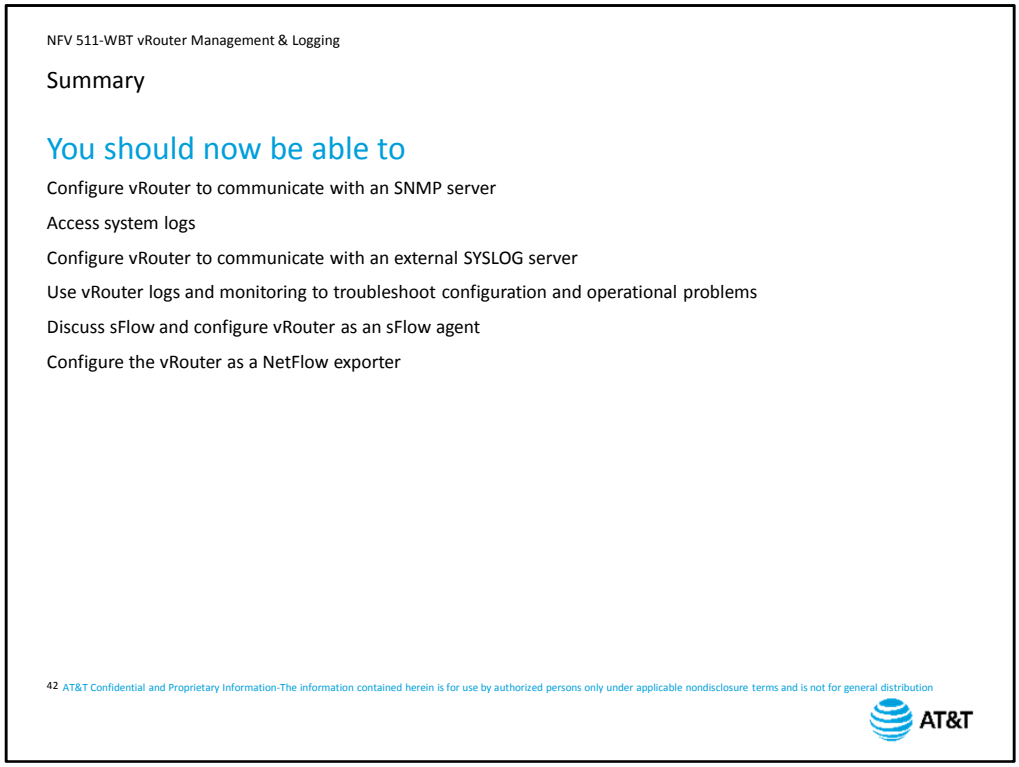

The most common application of custom logging is to send messages to an external syslog server (host). An external server may be collecting log information from multiple network devices, making it easier to correlate events.
In this scenario, we want to send *facility0* error level messages and above to the external syslog server at 192.168.51.150.



All the `log` commands we have shown you are "static" – that is, they access the log at the time you enter the command, but do not update your screen if new log entries occur.
If you want to view log entries in real time, you can use the `monitor` commands. Monitor puts your screen into a "wait and see" mode, and new log messages are displayed as they are generated.
Note that this is the same `monitor` command we used to enable debug-level logging for OSPF. If we want to view OSPF packets in real time, first enable the logging of the packets you want to view, then enable monitoring for the protocol in general.
To exit this mode and return to the Operational prompt, enter *CTRL+C*.



To review, the process for capturing debug-level messages is as follows.
First, enable logging for the appropriate protocol or feature. Remember to be as specific as possible – choose a particular rule for security, or a particular message or packet type for protocols.
Next, if you want to view the output in real time, enable monitoring for the protocol or feature.
Next, run your test traffic, or trigger a protocol update, or whatever you need to do to generate traffic relevant to your troubleshooting.
If you are not viewing it in real time, you can view it now in the system log using the show commands we discussed earlier.
When you are done, remember to go back and disable the logging.



This concludes the AT&T Vyatta 5600 vRouter Management and Logging course.
You should now be able to:

- Configure the vRouter to communicate with a SNMP server
- Access system logs
- Configure the vRouter to communicate with an external SYSLOG server
- Use vRouter logs and monitoring to troubleshoot configuration and operational problems
- Discuss sFlow and configure vRouter as an sFlow agent
- Configure the vRouter as a NetFlow exporter

We hope that this information has been useful, and that you will take additional AT&T University course in the future.
Thank you.

NFV 511-WBT vRouter Management & Logging

End of Course – vRouter Management & Logging