U

undebug

To turn off a previously set **debug** command, use the **undebug** privileged EXEC command. Use the **no** form of this command to disable the debug function.

undebug

Syntax Description

This command has no keywords or arguments.

Command Mode

Privileged EXEC.

Usage Guidelines

All debug commands are entered while in privilege EXEC mode, and most debug commands do not take any arguments. To enable the **debug atm rm** command, enter the following:

debug atm rm

To turn off the **debug atm rm** command, enter either the **no** form of this command or the **undebug** form of the command.

undebug debug atm rm

Related Commands

debug atm oam-all
debug atm oam-pkt
debug atm pnni
debug atm rm
debug atm sig
debug sscop

undelete

To recover a deleted file on a specified device, use the **undelete** EXEC command.

undelete *index* [*device***:**]

Syntax Description

index Number that indexes the file in the **dir**.

*device***:** (Optional) Device to contain the recovered configuration file. The colon (**:**) is required. Valid devices are as follows:

- **bootflash:** This device is the internal Flash memory.
- **slot0:** This device is the first PCMCIA slot on the ASP card.
- **slot1:** This device is the second PCMCIA slot on the ASP card.

Default

The default device is the one specified by the **cd** command.

Command Mode

EXEC.

Usage Guidelines

When you delete a file, the switch simply marks the file as deleted but does not erase the file. This command allows you to recover a "deleted" file on a specified Flash memory device. You must undelete a file by its index because you could have multiple deleted files with the same name. For example, the "deleted" list could contain multiple configuration files with the name *switch-config*. You undelete by index to indicate which of the many *switch-config* files from the list to undelete. Use the **dir** command to learn the index number of the file you want to undelete.

You cannot undelete a file if a valid (undeleted) one with the same name exists. Instead, you first delete the existing file and then undelete the file you want. For example, if you had an undeleted version of the *switch-config* file and you wanted to use a previous, deleted version, you could not simply undelete the previous version by index. First delete the existing *switch-config* file, and then undelete the previous *switch-config* file by index. You can delete and undelete a file up to 15 times.

If you try to recover the configuration file pointed to by the *config_file* environment variable, the system prompts you to confirm recovery of the file. This prompt reminds you that the *config_file* environment variable points to an undeleted file. To permanently delete all "deleted" files on a Flash memory device, use the **squeeze** command. If you try to recover a file that has the same name as an existing valid file, the system displays an error message.

Example

The following example recovers the deleted file whose index number is 1 to the Flash memory card inserted in slot 0 of the ASP card.

```
Switch# undelete 1 slot0:
```

Related Commands

delete
dir
squeeze

username

To establish a username-based authentication system at login, even though your network cannot support a TACACS service, use the **username** global configuration command.

username *name* [**nopassword** | **password** *encryption-type* **password** *password*]
username *name* **password** *secret*
username *name* [**access-class** *number*]
username *name* [**autocommand** *command*]
username *name* [**noescape**] [**nohangup**]

Syntax Description

name — Host name, server name, user ID, or command name. The *name* argument can only be one word. White spaces and quotation marks are not allowed.

nopassword — (Optional) No password is required for this user to log in. This is usually most useful in combination with the **autocommand** keyword.

password — (Optional) Specifies a possibly encrypted password for this username.

encryption-type — (Optional) A single-digit number that defines whether the text immediately following is encrypted and if so, what type of encryption is used. Currently defined encryption types are 0, which means that the text immediately following is not encrypted, and 7, which means that the text is encrypted using an encryption algorithm defined by Cisco.

password — (Optional) A password can contain embedded spaces and must be the last option specified in the **username** command.

secret — For CHAP authentication; specifies the secret for the local switch or the remote device. The secret is encrypted when it is stored on the local switch. This prevents the secret from being stolen. The secret can consist of any string of up to 11 printable ASCII characters. There is no limit to the number of username/password combinations that can be specified, allowing any number of remote devices to be authenticated.

access-class — (Optional) Specifies an outgoing access list that overrides the access list specified in the **access-class** line configuration command. It is used for the duration of the user’s session.

number — (Optional) The access list number.

autocommand — (Optional) Causes the specified command to be issued automatically after the user logs in. When the command is complete, the session is terminated. Because the command can be any length and contain imbedded spaces, commands using the **autocommand** keyword must be the last option on the line.

command — (Optional) The command string.

noescape — (Optional) Prevents a user from using an escape character on the host to which that user is connected.

nohangup — (Optional) Prevents the communication server from disconnecting the user after an automatic command (set up with the **autocommand** keyword) is complete. Instead, the user gets another login prompt.

Default

None.

Command Mode

Global configuration.

Usage Guidelines

The **username** command provides username/password authentication for login purposes only. (Note that it does not provide username/password authentication for enable mode when the **enable use-tacacs** command is also used.)

Multiple **username** commands can be used to specify options for a single user.

Add a **username** entry for each remote system that the local switch communicates with and requires authentication from. The remote device must have a **username** entry for the local switch. This entry must have the same password as the local switch's entry for that remote device.

This command can be useful for defining usernames that get special treatment, for example, an "info" username that does not require a password but connects the user to a general-purpose information service.

The **username** command is also required as part of the configuration for the Challenge Handshake Authentication Protocol (CHAP). For each remote system that the local switch communicates with from which it requires authentication, add a **username** entry.

Note To enable the local switch to respond to remote CHAP challenges, one **username** *name* entry must be the same as the **hostname** *name* entry that was already assigned to your switch.

If there is no *secret* specified and **debug serial-interface** is enabled, an error is displayed when an interface is established and the CHAP challenge is not implemented. Debugging information on CHAP is available using the **debug serial-interface** and **debug serial-packet** commands. For more information about **debug** commands, refer to the *Debug Command Reference* publication.

Examples

To implement a service similar to the UNIX **who** command, which can be entered at the login prompt and lists the current users of the switch, the **username** command takes the following form.

```
Switch# username who nopassword nohangup autocommand show users
```

To implement an information service that does not require a password to be used, the command takes the following form.

```
Switch# username info nopassword noescape autocommand telnet nic.ddn.mil
```

To implement an ID that works even if the TACACS servers all go down, the command takes the following form.

```
Switch# username superuser password superpassword
```

The following example configuration enables CHAP on interface serial 0. It also defines a password for the local server, Adam, and a remote server, Eve.

```
Switch# hostname Adam
Switch# interface serial 0
Switch# encapsulation ppp
Switch# ppp authentication chap
Switch# Switch# username Adam password oursystem
Switch# username Eve password theirsystem
```

When you look at your configuration file, the passwords are encrypted and the display looks similar to the following output.

```
Switch# hostname Adam
Switch# interface serial 0
encapsulation ppp
Switch# Switch# ppp authentication chap
Switch# username Adam password 7 1514040356
Switch# username Eve password 7 121F0A18
```

Related Command

hostname