

CONTENT FILTERING FOR BRANCH SRX SERIES AND J SERIES

Configuring Content Filtering on Branch SRX Series Services Gateways and J Series Services Routers

Table of Contents

Introduction	1
Scope.....	1
Design Considerations	1
Hardware Requirements	1
Software Requirements	1
Description and Deployment Scenario	1
Feature Description	1
Configuration.....	2
Deployment Scenarios	4
FTP Command Filtering	4
Using Multiple UTM Profiles	5
HTTP ActiveX Protection.....	7
Monitoring	8
Summary	8
About Juniper Networks.....	8

Table of Figures

Figure 1: UTM policies	2
Figure 2: UTM policies and feature profiles	2
Figure 3: FTP command filter example.....	4
Figure 4: Multiple UTM policies	5
Figure 5: HTTP filtering.....	7

Introduction

In recent years, applications have begun to encapsulate network protocols using Hypertext Transfer Protocol (HTTP). Several reasons explain this trend—in particular, the fact that HTTP traffic is commonly allowed through firewalls and proxy servers, while tunneled protocols are usually blocked. Because HTTP is commonly permitted, embedded applications are able to operate even when layer 3 and layer 4 firewalls are configured to block the underlying protocols. Although HTTP encapsulation is the most common example, protocols may be tunneled in other generally allowed protocols. As such, firewalls now have to inspect higher layers of the protocol stack.

Scope

Juniper Networks® Junos® operating system release 9.5, adds UTM support for Juniper Networks J Series Routers and selected Juniper Networks SRX Series Services Gateways. Content filtering, one of several features, including antivirus, anti-spam and web filtering, that comprise Juniper's UTM suite, provides the ability to allow or block traffic based on the content carried by the underlying protocol.

Design Considerations

When deciding to deploy content filtering, network designers should consider the performance impact of value added security. Product guidelines can be found on J Series Service Routers and SRX Series Services Gateways datasheets.

Hardware Requirements

- SRX Series Services Gateways for the branch including the SRX100, SRX210, SRX240 and SRX650 (content filtering is not available on high-end SRX Series platforms)
- J Series Services Routers including the J2320, J2350, J4350, and J6350 Services Routers

Software Requirements

- Junos OS release 9.5 or later for all supported SRX Series Services Gateways (content filtering is not available on high-end SRX Series platforms)
- Junos OS version 9.5 or later on J Series Services Routers

Description and Deployment Scenario

Feature Description

Content filtering enables traffic to be permitted or blocked based on inspection of the following parameters:

- MIME pattern
- File extension
- Protocol command

This feature is supported for HTTP, FTP, and mail protocols such as Simple Mail Transfer Protocol (SMTP), Post Office Protocol 3 (POP3), and IMAP. In addition, HTTP traffic may also be permitted or denied based on:

- ActiveX
- Java Applet
- Cookies
- EXE file
- ZIP file

Once content is blocked, users are notified in one of two ways. A message embedded in the protocol (that is, an FTP, HTTP, or mail error message) may be sent to the user; or, for email, a message can be sent to either the sender and/or receiver of the mail. In both cases, the content of the message is configurable by the administrator.

As of Junos OS version 9.5, entries in the allow/block list are compared using string matching only. Character expansion is not available, so no wildcard characters are supported. Character comparison is case insensitive; both uppercase and lowercase characters are treated as if they were the same.

Both MIME and protocol command filtering provide two lists; one for allowed and one for blocked content. Content in the allowed list will be forwarded even if it also matches a blocked list entry, which is particularly useful when creating MIME filters, as one can deny whole categories (applications) while allowing more specific content (application-MS Word). If content is not specifically denied, it will be forwarded.

Configuration

The unified threat management (UTM) implementation in Junos OS leverages security policies as a central point where traffic is classified and directed to the appropriate modules for processing. In practice, a UTM policy specifying all UTM-related parameters is attached to a security policy, and matching traffic is processed by the UTM module according to the configuration of the UTM policy.

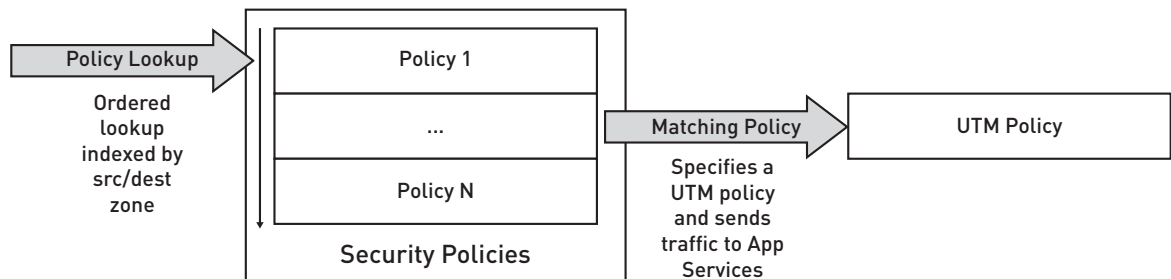


Figure 1: UTM policies

In a similar fashion, a UTM policy ties a set of protocols to one or multiple feature profiles. Each feature profile determines the specific configuration for each feature (antivirus, content filtering, anti-spam). In this document, we will only present the content filtering feature; so the UTM policies shown in the examples only reference content filtering profiles.

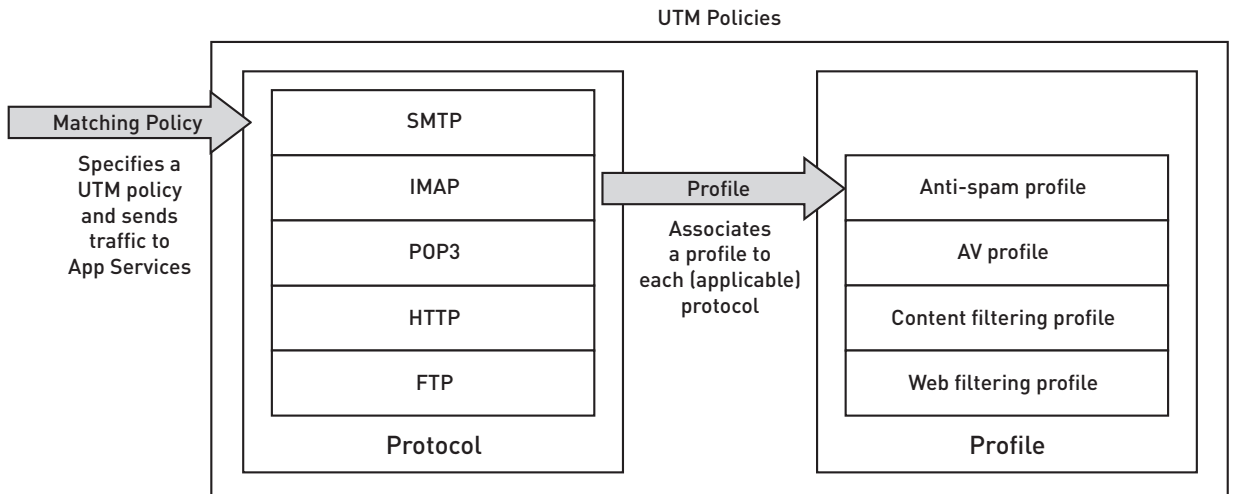


Figure 2: UTM policies and feature profiles

The content filtering configuration hierarchy is found under [security utm feature-profile] and is shown below:

```
.....
content-filtering {
  profile <name> {
    permit-command <cmd-list>;
    block-command <cmd-list>;
    block-extension <file-ext-list>;
    block-mime {
      list <mime-list>;
      exception <ex-mime-list>;
    }
    block-content-type {
      activex;
      java-applet;
      exe;
      zip;
      http-cookie;
    }
    notification-options {
      type { protocol-only | message };
      not-notify-mail-sender;
      custom-message <msg>;
    }
  }
}
.....
```

In order to provide a simple and reusable way to identify objects, content filtering allows for the creation of pattern lists that can be referenced multiple times by different configuration profiles. The MIME, extension, and command list are configured under the [security utm custom-objects] hierarchy and consist of a list of one or more strings with no wildcard characters.

```
.....
custom-objects {
  protocol-command <name> {
    value [<list of commands>];
  }
  mime-pattern <name> {
    value [<list of commands>];
  }
  filename-extension <name> {
    value [<list of commands>];
  }
}
.....
```

Deployment Scenarios

The following section introduces some common deployment scenarios and provides their associated configurations.

FTP Command Filtering

In our first example, let's start with the simple network shown below:

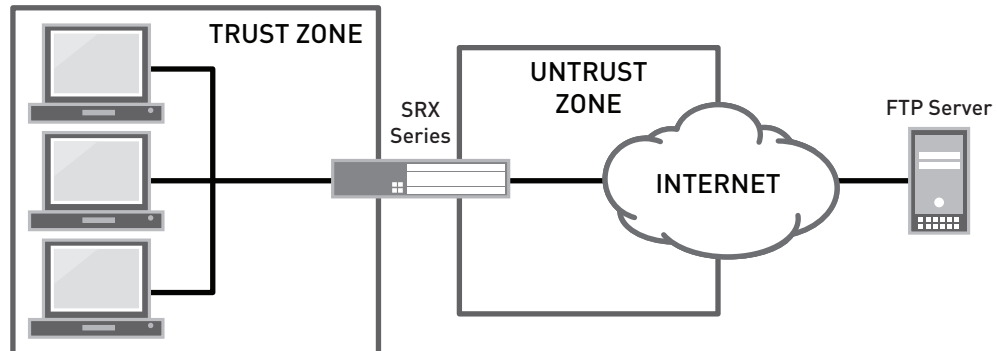


Figure 3: FTP command filter example

In this configuration, we will prevent users in the trust zone from uploading files to an FTP server by blocking the STOR command. We start by defining a protocol command list and reference it from the content filtering profile.

```

.....
security {
    utm {
    custom-objects {
        protocol-command {
            ftp-put {
                value STOR;
            }
        }
    }
}
feature-profile {
    content-filtering {
        profile block-ftp-upload {
            block-command ftp-put;
        }
    }
}
}
}
.....

```

Once the content filtering profile is defined, it can be used to create a UTM policy that maps the profile to FTP traffic. For FTP traffic (and only for FTP traffic), traffic direction is important, as different profiles can be applied in each direction (from client to server or vice versa).

```

.....
security {
    utm {
    utm-policy ftp-inspect {
        content-filtering {
            ftp {
                upload-profile block-ftp-upload;
            }
        }
    }
}
}
}
.....

```

The final step is to select which traffic will be sent to the UTM module for processing by configuring a security policy.

```

security {
  policies {
    from-zone trust to-zone untrust {
      policy Inspect-Internet-Traffic {
        match {
          source-address any;
          destination-address any;
          application any;
        }
        then {
          permit {
            application-services {
              utm-policy ftp-inspect;
            }
          }
        }
      }
    }
  }
}

```

Using Multiple UTM Profiles

Now let's assume that the network has two zones that require different policies. Traffic from the trust to the untrust zone will still use the no-upload profile. Traffic from the finance zone will not only prevent file uploads, but will block the download of executable files using extension-based filtering.

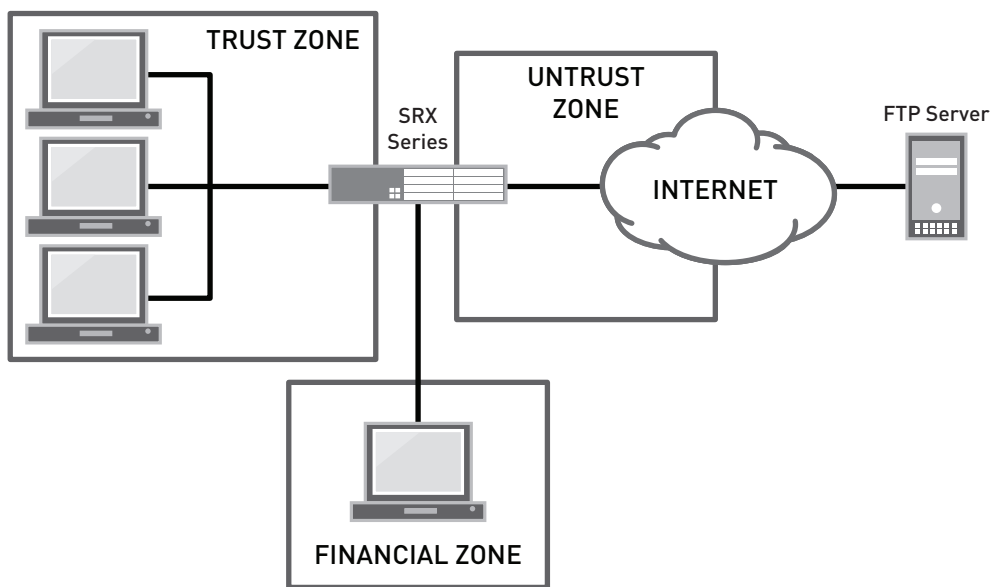


Figure 4: Multiple UTM policies

New UTM and content filtering profiles will be created to block files with common executable extensions (the executable list shown is by no means exhaustive and is only meant to serve as an example).

```

utm {
  custom-objects {
    filename-extension {
      executables {
        value [ ADE ADP BAS BAT CHM CMD COM CPL CRT DLL DOT EXE HLP HTA INF INS ISP JS

```

```

JSE LNK MDB MDE MSC MSI MSP MST OCX PCD PIF POT PPT REG SCR SCT SHB SHS SYS URL VB VBE VBS WSC
WSF WSH XLT ];
    }
  }
}
feature-profile {
  content-filtering {
    profile block-exe {
      block-extension executables;
    }
  }
}
utm-policy ftp-block-exe {
  content-filtering {
    ftp {
      upload-profile block-exe;
      download-profile block-exe;
    }
  }
}
}

```

Just like in our previous example, we will apply the ftp-inspect policy to traffic from the trust to the untrust zone. In addition, the newly created ftp-block-exe policy will be applied to traffic between the finance and untrust zones.

```

policies {
  from-zone trust to-zone untrust {
    policy Inspect-Internet-Traffic {
      match {
        source-address any;
        destination-address any;
        application any;
      }
      then {
        permit {
          application-services {
            utm-policy ftp-inspect;
          }
        }
      }
    }
  }
  from-zone finance to-zone untrust {
    policy allow-all-block-executables {
      match {
        source-address any;
        destination-address any;
        application any;
      }
      then {
        permit {
          application-services {
            utm-policy ftp-block-exe;
          }
        }
      }
    }
  }
}

```

HTTP ActiveX Protection

Since ActiveX and Java components can be used to execute malicious code on a client machine, it is sometimes desirable to prevent users from accessing sites that use them. In the following configuration example, ActiveX and Java will be blocked, and an error message page will be generated indicating that the policy has blocked the offending components.

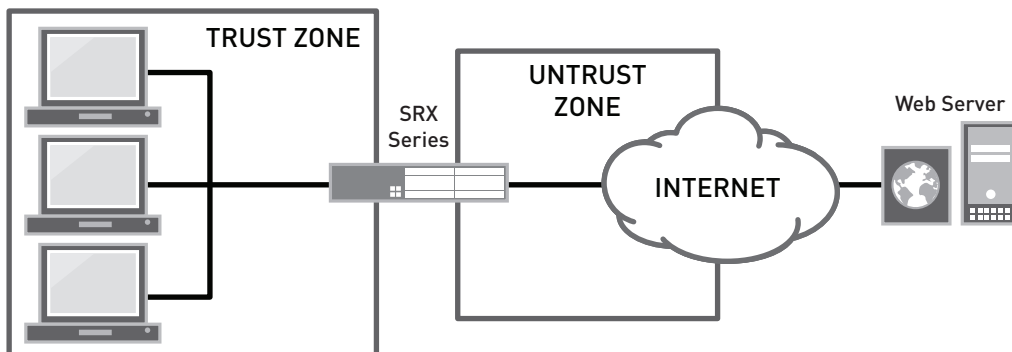


Figure 5: HTTP filtering

```

policies {
  from-zone trust to-zone untrust {
    policy block-http-components {
      match {
        source-address any;
        destination-address any;
        application any;
      }
      then {
        permit {
          application-services {
            utm-policy block-http-components;
          }
        }
      }
    }
  }
  policy-rematch;
}
utm {
  feature-profile {
    content-filtering {
      profile block-http-components {
        block-content-type {
          activex;
          java-applet;
        }
        notification-options {
          custom-message "You are not allowed to view pages that use ActiveX or Java
components.";
        }
      }
    }
  }
  utm-policy block-http-components {
    content-filtering {
      http-profile block-http-components;
    }
  }
}

```

Monitoring

To view a summary of what the content filtering engine has blocked, issue the show security utm content-filtering statistics command.

```
show security utm content-filtering statistics
```

```
Content-filtering-statistic:          Blocked
Base on command list:                1
Base on mime list:                   0
Base on extension list:              0
ActiveX plugin:                      0
Java applet:                         0
EXE files:                           0
ZIP files:                           0
HTTP cookie:                         2
```

It is also sometimes useful to verify that traffic is indeed being processed by the correct security policy (where the UTM profile is applied). The show security flow session command looks into the session table and verifies which policy is processing specific traffic.

```
show security flow session
```

```
Session ID: 991, Policy name: block-http-components/4, Timeout: 1798
In: 10.1.1.11/60697 --> 74.125.19.103/80;tcp, If: fe-0/0/5.0
Out: 74.125.19.103/80 --> 172.19.101.42/1127;tcp, If: fe-0/0/7.0
```

Summary

The content filtering feature introduced in Junos OS 9.5 provides a simple way to examine and control application-layer traffic forwarded through SRX Series Services Gateways or J Series Services Routers. This feature provides a foundation for data loss prevention, which is becoming increasingly important as data theft attacks are on the rise.

About Juniper Networks

Juniper Networks, Inc. is the leader in high-performance networking. Juniper offers a high-performance network infrastructure that creates a responsive and trusted environment for accelerating the deployment of services and applications over a single network. This fuels high-performance businesses. Additional information can be found at www.juniper.net.

Corporate and Sales Headquarters

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, CA 94089 USA
Phone: 888.JUNIPER (888.586.4737)
or 408.745.2000
Fax: 408.745.2100
www.juniper.net

APAC Headquarters

Juniper Networks (Hong Kong)
26/F, Cityplaza One
1111 King's Road
Taikoo Shing, Hong Kong
Phone: 852.2332.3636
Fax: 852.2574.7803

EMEA Headquarters

Juniper Networks Ireland
Airside Business Park
Swords, County Dublin, Ireland
Phone: 35.31.8903.600
EMEA Sales: 00800.4586.4737
Fax: 35.31.8903.601

To purchase Juniper Networks solutions, please contact your Juniper Networks representative at 1-866-298-6428 or authorized reseller.

Copyright 2010 Juniper Networks, Inc. All rights reserved. Juniper Networks, the Juniper Networks logo, Junos, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. All other trademarks, service marks, registered marks, or registered service marks are the property of their respective owners. Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.