digitalscepter

San Diego COE Palo Alto Networks Workshop

July 16th, 2026

Agenda

1. SSL Decryption Overview and Best Practices
2. Benefits of EDLs
3. User-ID Best Practices
4. App-ID Best Practices

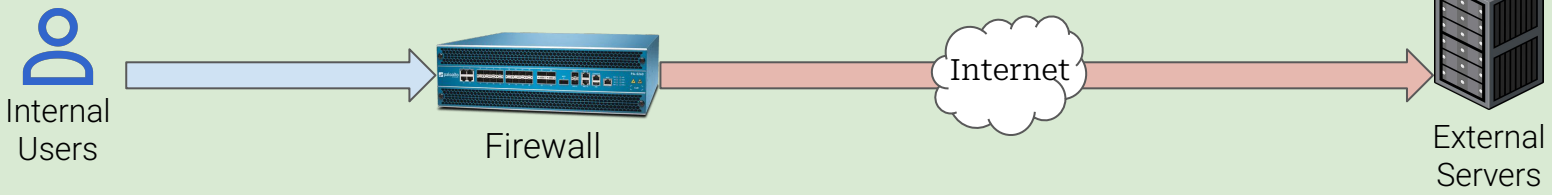
SSL Decryption and Best Practices

Encrypted Traffic and the Visibility Gap

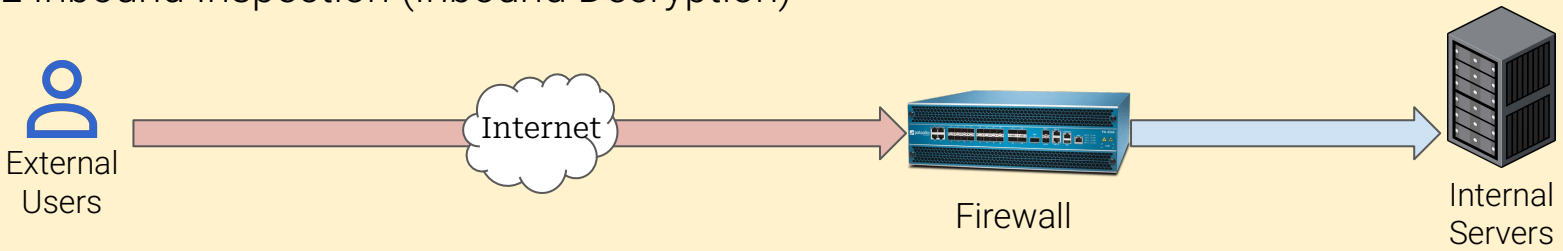
- Most modern web and application traffic uses TLS encryption
- Encryption protects legitimate users—but also conceals malware, exploits, command-and-control traffic, data exfiltration, etc.
- Without decryption, the firewall can inspect connection metadata but not the encrypted payload
- Security profiles have limited effectiveness when content remains encrypted
- Decryption restores visibility so App-ID, Threat Prevention, URL Filtering, File Blocking, and WildFire can inspect the traffic

Types of Decryption

SSL Forward Proxy (Outbound Decryption)



SSL Inbound Inspection (Inbound Decryption)



digitalscepter

SSL Forward Proxy

- Protects users accessing external websites and applications
- The firewall proxies the connection between the internal client and external server
- Primarily used for outbound internet traffic
- Requires clients to trust the firewall's Forward Trust CA certificate

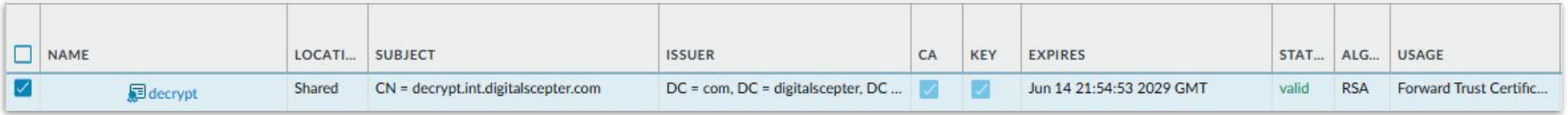
SSL Inbound Inspection

- Protects internally hosted servers from encrypted inbound attacks
- The firewall uses the internal server’s certificate and private key
- Primarily used for public-facing web servers and applications
- Does not require certificates to be installed on external clients

Certificate Chain and Trust Model

- TLS uses certificates to authenticate the identity of a server.
- A typical chain includes:
  - Root Certificate Authority
  - Intermediate Certificate Authority
  - Server certificate
- Clients trust certificates issued by CAs in their trusted root store.
- Forward Proxy inserts the firewall into the TLS trust relationship.
- The firewall validates the destination server and generates a substitute certificate for the client.
- Certificate deployment and PKI planning should occur before enabling decryption

Forward Trust Certificates



NAME	LOCATI...	SUBJECT	ISSUER	CA	KEY	EXPIRES	STAT...	ALG...	USAGE
decrypt	Shared	CN = decrypt.int.digitalscepter.com	DC = com, DC = digitalscepter, DC ...	✓	✓	Jun 14 21:54:53 2029 GMT	valid	RSA	Forward Trust Certific...

- Used when the destination server presents a certificate the firewall considers trusted
- Must be a CA certificate
- The issuing certificate must be trusted by managed client devices
- Normally distributed through Group Policy, MDM, Intune, or another endpoint-management platform

digitalscepter

Forward Untrust Certificates

NAME	LOCATI...	SUBJECT	ISSUER	CA	KEY	EXPIRES	STAT...	ALG...	USAGE
decrypt_untrust	vsys1	CN = UNTRUSTED-DECRYPT - DO NOT PROCEED	NTRUSTED-DECRYPT - D...	☑	☑	Jun 16 20:22:26 2027 GMT	valid	RSA	Forward Untrust Certi...

- Used when the destination presents an expired, self-signed, invalid, or otherwise untrusted certificate
- Should not be trusted by endpoint devices
- Allows users to continue receiving a certificate warning
- Prevents the firewall from making an untrusted destination appear trustworthy

How Palo Alto Networks Performs Decryption

- The firewall intercepts a TLS connection matching a Decryption policy rule
- It creates a TLS session with the destination server
- It validates the server certificate
- It dynamically generates a substitute server certificate for the client
- The client establishes a separate TLS session with the firewall
- Traffic is decrypted in memory and inspected
- Security policy and security profiles are applied
- Permitted traffic is re-encrypted and forwarded

SSL Forward Proxy Policies

				Source				Destination							
	NAME	LOCATION	TAGS	ZONE	ADDRESS	USER	DEVICE	ZONE	ADDRESS	DEVICE	URL CATEGORY	SERVICE	ACTION	TYPE	DECRYPTION
1	No Decrypt - Confid...	demo	none	inside	any	any	any	outside	any	any	financial-services government health-and-medicine shopping	any	no-decrypt	ssl-forward-proxy	none
2	Decrypt by User	demo	none	inside	any	ds\domain u...	any	outside	any	any	any	any	decrypt	ssl-forward-proxy	dp_standard
3	Decrypt by IP	demo	none	inside	10.20.0.0/16	any	any	outside	any	any	any	any	decrypt	ssl-forward-proxy	dp_standard

- Top rule should always prevent decryption on confidential URL Categories (URL Filtering license is required for this)
- Subsequent rules should enable decrypt by User, IP address, or both

digitalscepter

SSL Forward Proxy - Important Settings

- QUIC traffic should be blocked to force fallback to TLS which allows for decryption. QUIC traffic can't be decrypted

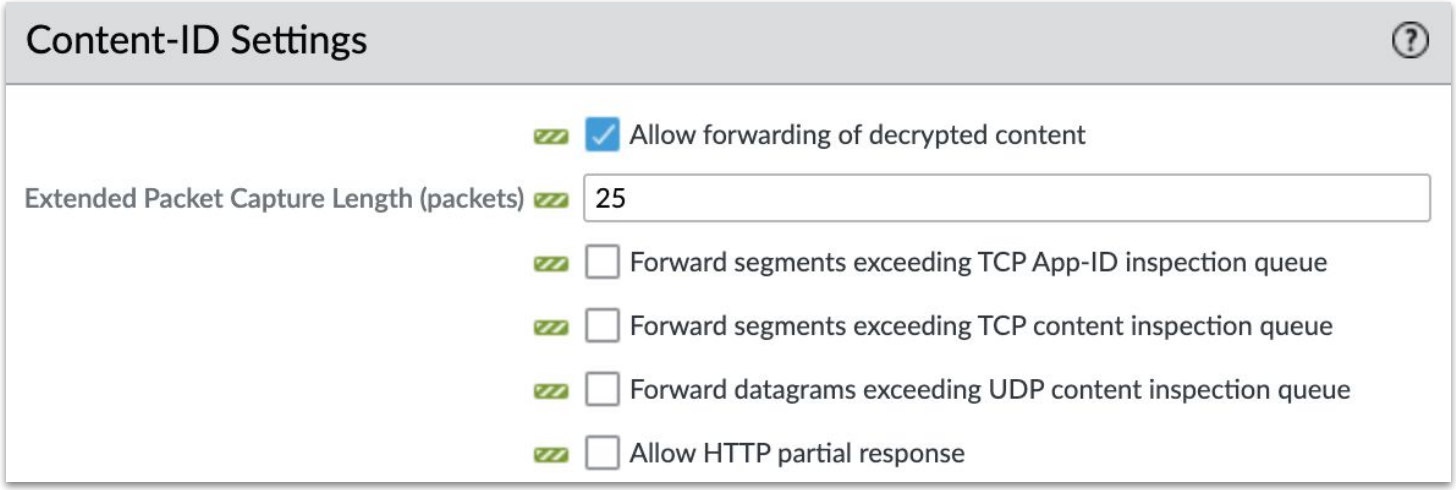
NAME	TAGS	ZONE	ADDRESS	USER	DEVICE	ZONE	ADDRESS	DEVICE	APPLICATION	SERVICE	ACTION
Block Quic	internet	any	INTERNAL_NETWORKS	any	any	outs...	any	any	quic	any	Deny

SSL Forward Proxy - Important Settings

- ECH (Encrypted Client Hello) can also prevent websites from being decrypted since the firewall will not see the HTTP GET request, meaning no decryption and no URL filtering logs, similar to with QUIC. ECH can be blocked via DNS Security or by filtering DNS replies on your DNS server

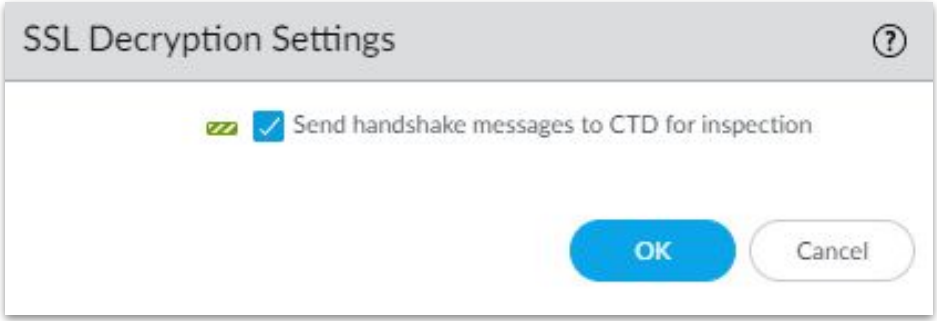
SSL Forward Proxy - Important Settings

- Decrypted files should be sent to WildFire
  - **Device** > **Setup** > **Content-ID** > **Content-ID Settings**



SSL Forward Proxy - Important Settings

- Enable inspection of SSL handshake messages
  - **Device** > **Setup** > **Session** > **SSL Decryption Settings**



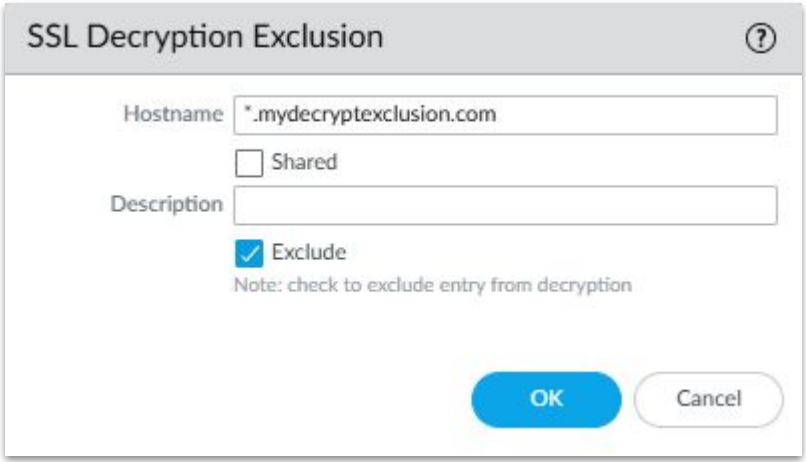
digitalscepter

SSL Forward Proxy - Unsupported Websites and Apps

- Decryption may fail when an application:
  - Uses certificate pinning
  - Requires mutual TLS or client-certificate authentication
  - Uses an unsupported cipher or TLS feature
  - Maintains its own certificate trust store
  - Uses a nonstandard implementation of TLS

SSL Forward Proxy - Exclusions

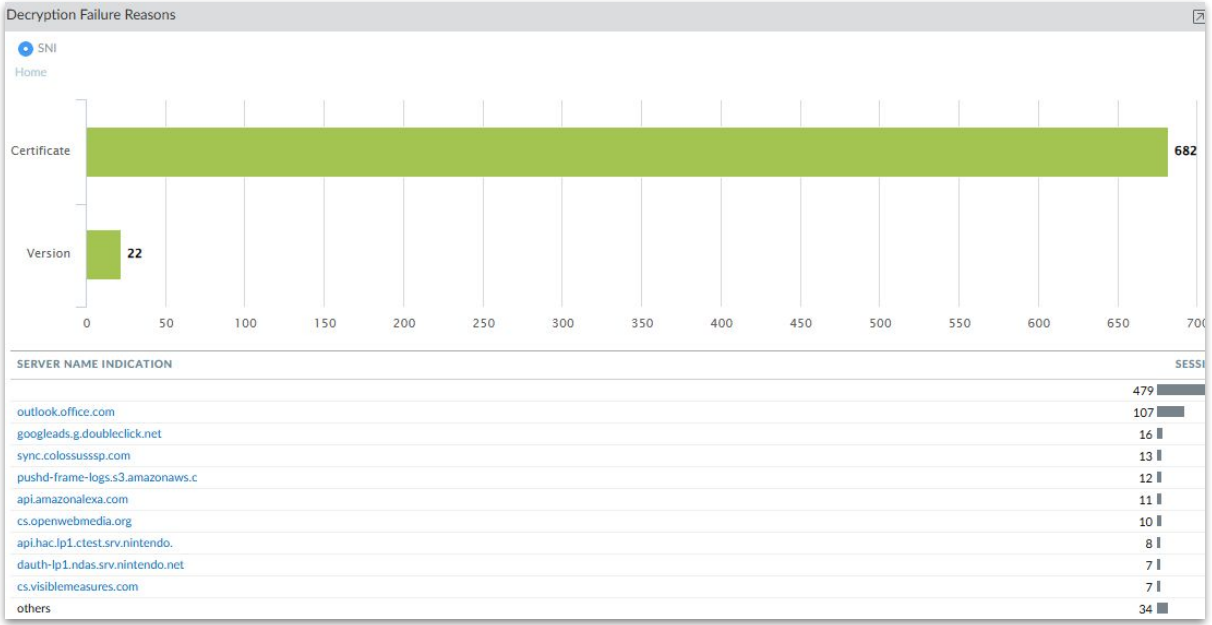
- Global exclusions can be configured at **Device** > **Certificate Management** > **SSL Decryption Exclusion**



digitalscepter

Troubleshooting Decryption Failures

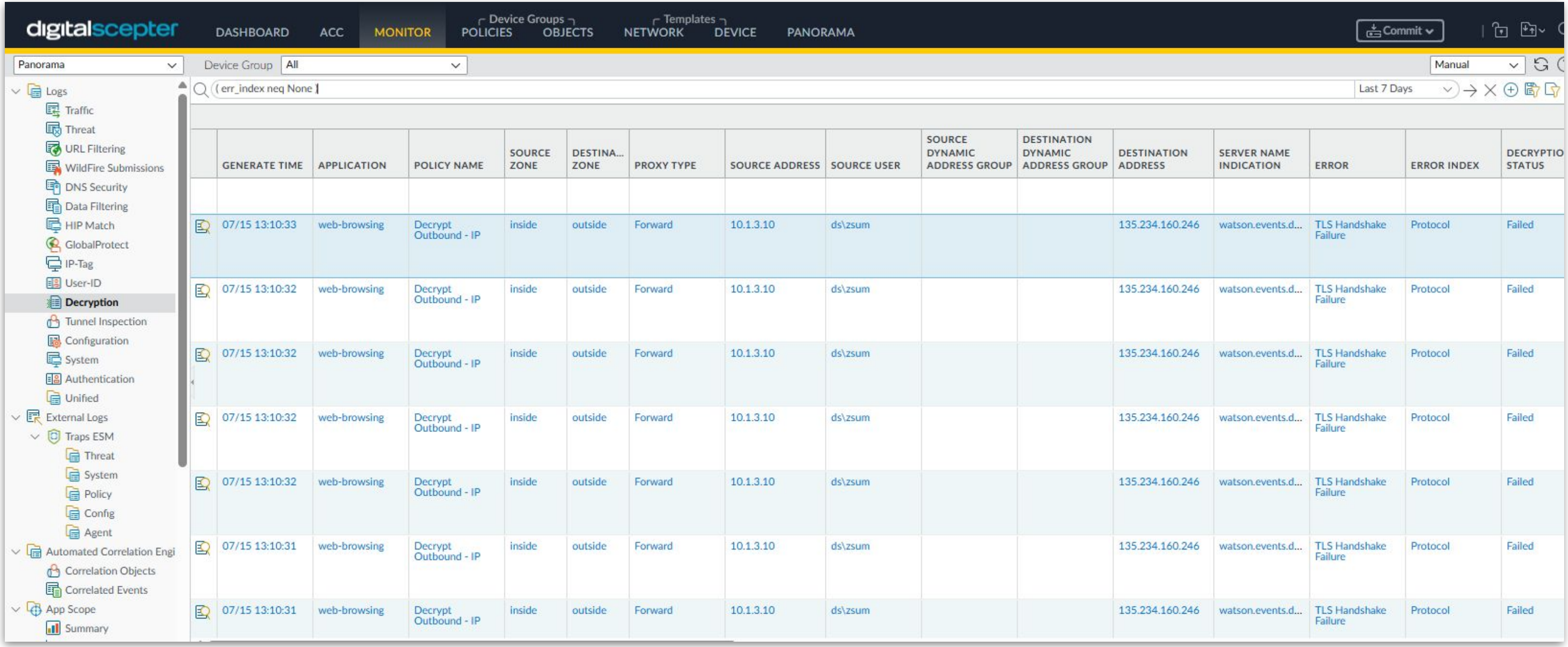
- **ACC** > **SSL Activity** > **Decryption Failure Reasons**
  - Can proactively check for domains failing to be decrypted
  - Choose to create exceptions, or leave them be if the domains are not required



digitalscepter

Troubleshooting Decryption Failures

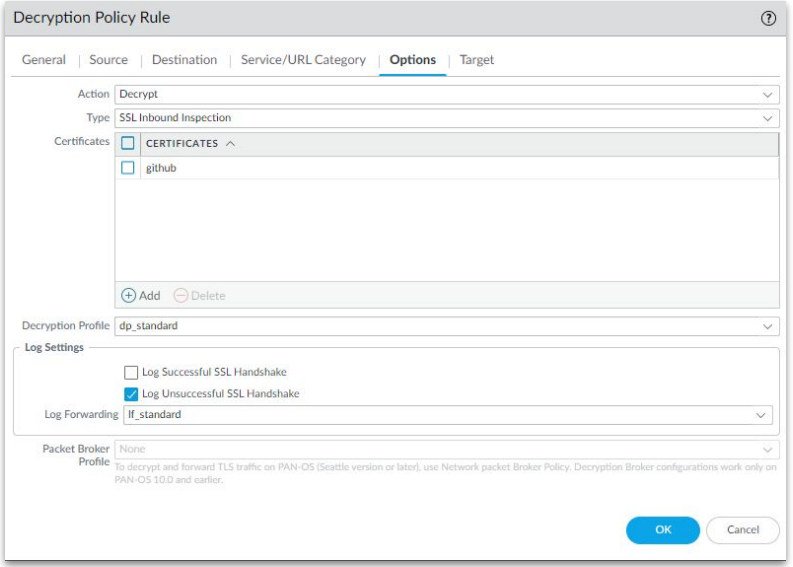
- **Monitor** > **Logs** > **Decryption**
  - Query: (err_index neq None)



SSL Inbound Inspection Policies

NAME	LOCATION	TAGS	Source				Destination			URL CATEGORY	SERVICE		
			ZONE	ADDRESS	USER	DEVICE	ZONE	ADDRESS	DEVICE			ACTION	TYPE
inspect github	labfw01	none	globalprote...	any	any	any	inside	10.1.131.35/32	any	any	any	decrypt	ssl-inbound-inspection

- Each rule will target a specific internal server
- The certificate (with private key) must be uploaded to the firewall and attached on the decrypt policy Options tab
- The certificate must be an exact match as the certificate running on the destination server



digitalscepter

SSL Inbound Inspection Logs

GENERATE TIME	TYPE	FROM ZONE	TO ZONE	SOURCE	SOURCE USER	DESTINATION	DECRYPTED	TO PORT	APPLICATION	ACTION	RULE	SESSION END REASON
03/31 17:13:57	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	443	websocket	allow	Allow Admins to Inside	tcp-fin
03/31 16:51:45	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	443	git-base	allow	Allow Admins to Inside	aged-out
03/31 16:51:34	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	443	github-base	allow	Allow Admins to Inside	aged-out
03/31 16:51:33	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	443	github-base	allow	Allow Admins to Inside	aged-out
03/31 16:46:28	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	443	github-base	allow	Allow Admins to Inside	aged-out
03/31 16:41:54	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	443	websocket	allow	Allow Admins to Inside	tcp-fin
03/31 16:41:23	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	443	github-base	allow	Allow Admins to Inside	aged-out
03/31 16:36:20	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	443	github-base	allow	Allow Admins to Inside	aged-out
03/31 16:31:17	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	443	github-base	allow	Allow Admins to Inside	aged-out
03/31 16:26:45	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	8443	github-base	allow	Allow Admins to Inside	aged-out
03/31 16:26:45	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	8443	github-base	allow	Allow Admins to Inside	aged-out
03/31 16:26:13	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	443	github-base	allow	Allow Admins to Inside	aged-out
03/31 16:21:19	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	443	git-base	allow	Allow Admins to Inside	aged-out
03/31 16:21:08	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	443	github-base	allow	Allow Admins to Inside	aged-out
03/31 16:21:07	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	443	github-base	allow	Allow Admins to Inside	aged-out
03/31 16:16:03	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	443	github-base	allow	Allow Admins to Inside	aged-out
03/31 16:10:59	end	vpn	inside	172.21.2.7	ds\zsum	10.1.131.35	yes	443	github-base	allow	Allow Admins to Inside	aged-out

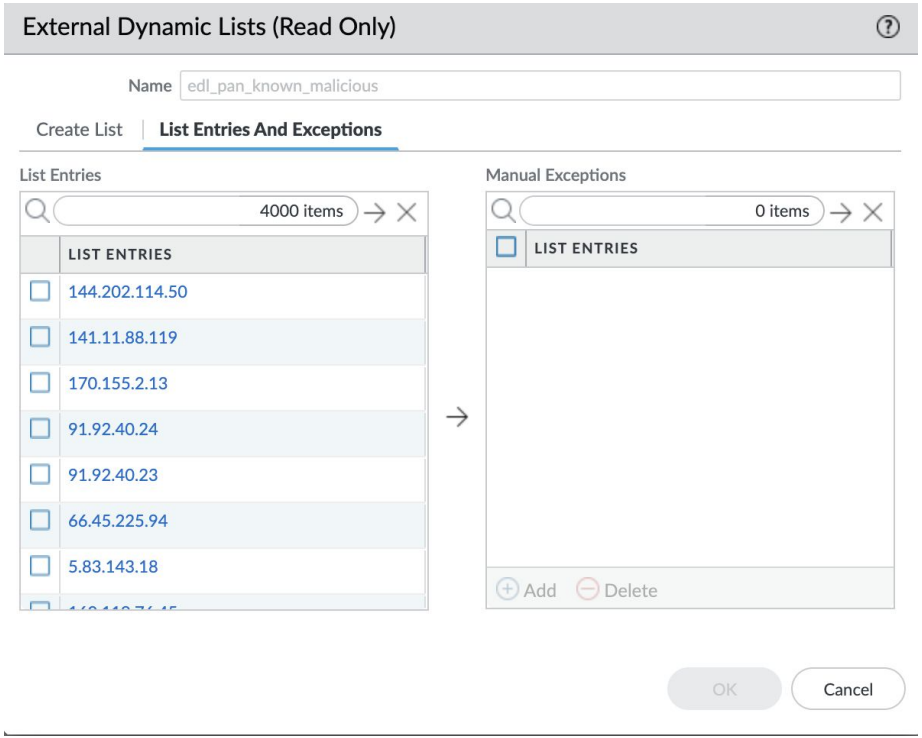
digitalscepter

Benefits of EDLs

What is an External Dynamic List (EDL)

- A text-based list hosted on a web server and periodically retrieved by the firewall
- Allows policy to reference EDLs that change without manually editing firewall configuration
- EDL types include IP addresses, domains and URLs with each type being usable in different capacities across your configuration
- The firewall applies changes after retrieving the updated list without requiring another configuration commit

What is an External Dynamic List (EDL)



digitalscepter

Why Use EDLs?

- Reduce manual address object and policy maintenance
- Respond more quickly to newly discovered threats
- Automate policy updates using trusted external data
- Separate changing data from the firewall configuration
- Allow security operations teams to update enforcement without receiving full firewall administrative access
- Improve consistency across multiple firewalls and locations
- Reduce the need for emergency commits
- Support both security blocking and business allowlisting

How EDLs Work

- An administrator creates an EDL object on the firewall or Panorama
- The EDL object references a URL containing the source list
- The firewall retrieves the list using its management interface or configured service route
- Entries are validated against the configured list type, e.g. valid IP addresses for an IP Address list type
- Valid entries are loaded into the firewall
- Security policy or a security profile references the EDL object
- The firewall retrieves updates at the configured interval, as short as every 5 minutes
- Updated entries begin affecting policy without changing the policy rule or performing a commit

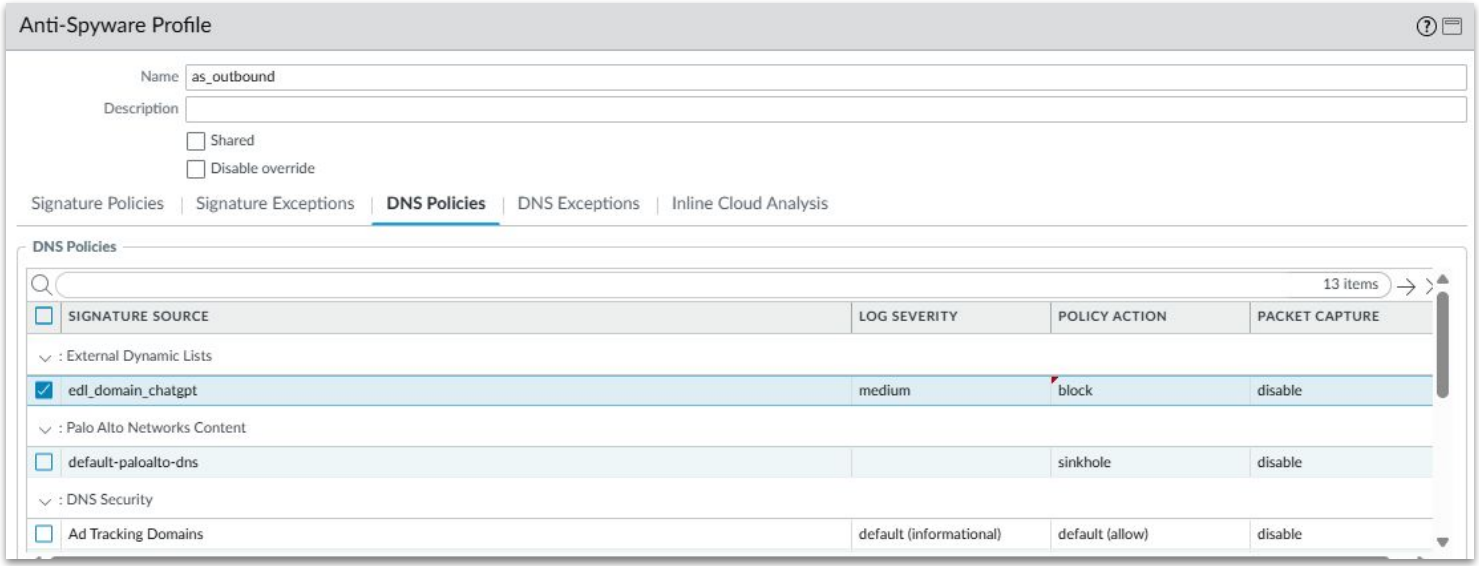
IP Address Lists

- Individual IPv4 or IPv6 addresses
- IP subnets
- IP address ranges
- Commonly used as source address or destination address match criteria in security and decryption policies

					Source			Destination					
	NAME	LOCATION	TAGS	TYPE	ZONE	ADDRESS	USER	DEVICE	ZONE	ADDRESS	DEVICE	APPLICATION	SERVICE	ACTION
3	Block Bad IPs Inbound	global	Inbound	universal	outside	Palo Alto Networks - Bulletproof IP addresses Palo Alto Networks - High risk IP addresses Palo Alto Networks - Known malicious IP ad... Palo Alto Networks - Tor exit IP addresses	any	any	any	any	any	any	any	Drop
4	Block Bad IPs Outbo...	global	Internet access	universal	any	any	any	any	outside	Palo Alto Netw... Palo Alto Netw... Palo Alto Netw... Palo Alto Netw...	any	any	any	Drop

Domain Lists

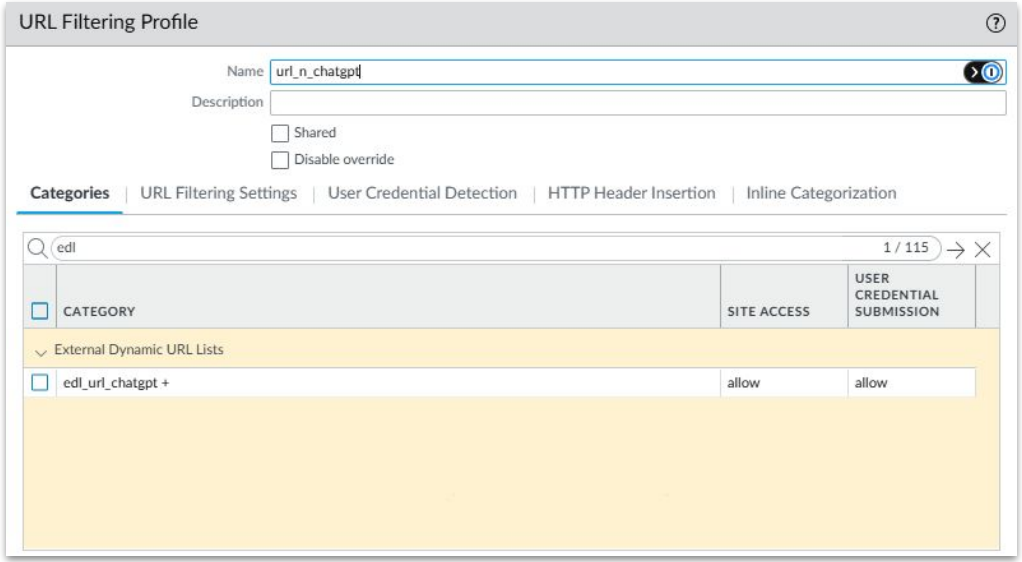
- Domain names associated with malicious or prohibited destinations
- Commonly used with DNS Security and DNS sinkholing

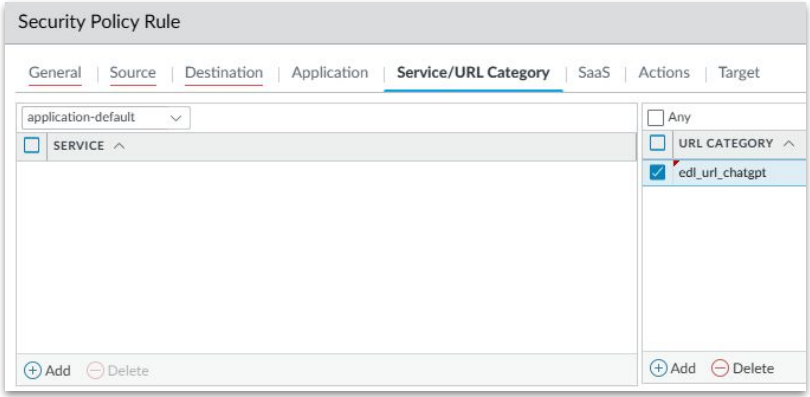


digitalscepter

URL Lists

- Websites, domains or specific URL paths
- Commonly used in URL Filtering profiles and Security policy





digitalscepter

Palo Alto Networks Built-In EDLs

- Maintained by Palo Alto Networks
- Delivered through content updates rather than a customer-hosted web server
- Include:
  - Known malicious IP addresses
  - High-risk IP addresses
  - Bulletproof hosting provider IP addresses
  - Tor exit node IP addresses
- Require an active Threat Prevention license
- Cannot be directly edited or deleted
- Can be referenced directly in policy
- Can also be used as the source for another EDL when exceptions are needed

Threat Intelligence Use Cases

- Import indicators from a SIEM, SOAR or threat intelligence platform
- Apply intelligence from industry information-sharing groups, e.g. MS-ISAC
- Enforce temporary containment while permanent controls are developed
- Automatically remove indicators when they are no longer present in the source list

Business/Allowlist Use Cases

- SSL Decryption inclusions/exclusions
- Allow access to approved partner IP addresses
- Maintain vendor and SaaS allowlists
- Allow 3rd party trusted vulnerability scanners
- Allow approved cloud service endpoints
- Restrict administrative access to authorized source addresses
- Publish lists from a central internal system for multiple firewalls

Hosting an Internal EDL

- Host the list on a reliable HTTPS server
  - Avoid HTTP as it can be MITM’d
- Restrict who can modify the source file
- Use version control or change logging where possible
- Make the server reachable from every firewall that consumes the list
- Design for redundancy when the EDL supports critical enforcement

Formatting an EDL

- Use one entry per line
- Include only entries matching the selected list type, e.g. no URL’s in an IP list
- Do not add http:// or https:// to URL entries
- Remove blank or malformed entries
- Use comments only where supported
- Validate automatically generated lists before publishing
- Keep the file simple and avoid HTML formatting, a regular text file is all you want

Refresh Intervals and Updates

- Hourly is the default retrieval interval
- Available options can include every five minutes, hourly, daily, weekly or monthly
- Choose the interval based on how quickly the data changes
- Use shorter intervals for active threat or incident-response feeds
- Use longer intervals for stable business allowlists
- Excessively frequent retrieval may create unnecessary load
- Administrators can use ‘Import Now’ button to trigger an immediate retrieval
- Remember, no commits are required for retrieved lists to take effect

EDL Tidbits

- Each platform has limits on how many IP, URL, and Domain entries it can maintain
- EDL's may have false positives, so Palo Alto allows exclusions to be configured within EDL's. These should be monitored and removed when appropriate as there are exclusion limits as well
- If an EDL becomes inaccessible the firewall will cache and use the last retrieved EDL list

User-ID Best Practices

What is User-ID

- The process of mapping network activity to usernames versus relying only on source IP addresses
- Adds identity context to Security policy, logs, and reporting
- Allows administrators to control application access based on who the user is
- Combines IP-to-user mapping with directory group information
- Supports internal, remote, wireless and shared-system users
- Included as a standard capability of Palo Alto Networks firewalls

What is User-ID

- The process of mapping network activity to usernames versus relying only on source IP addresses
- Adds identity context to Security policy, logs, reports and the Application Command Center
- Allows administrators to control application access based on who the user is
- Combines IP-to-user mapping with user-to-group mapping
- Supports users on internal networks, remote networks, wireless networks and shared systems
- Included as a standard capability of Palo Alto Networks firewalls

Why User-ID Based Policies Matter

- IP addresses change through DHCP, VPN connections and wireless roaming
- Multiple users may use the same device at different times
- A single user may connect from several devices
- IP-based provide little visibility in auditing network usage
- User-based rules align access with business roles
- Identity improves investigations, reporting and accountability

Why User-ID Based Policies Matter

- IP addresses change through DHCP, VPN connections and wireless roaming
- Multiple users may use the same device at different times
- A single user may connect from several devices
- IP-based provide little visibility in auditing network usage
- User-based rules align access with business roles
- Identity improves investigations, reporting and accountability

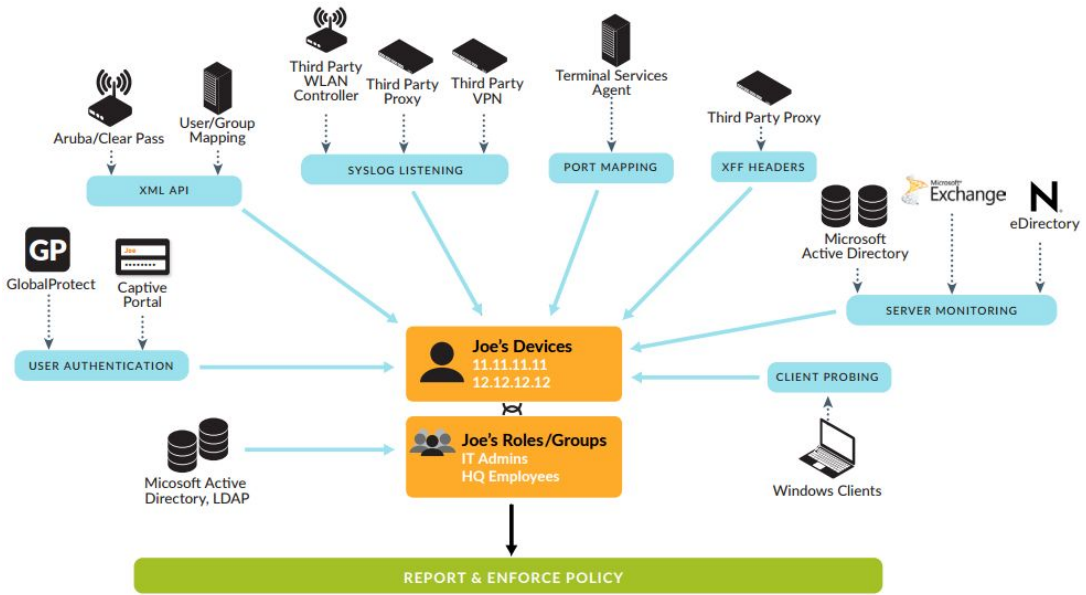
User Mapping vs Group Mapping

- User Mapping
  - Answers: Who is currently using this IP address?
  - Built from login events, VPN authentication, syslog or API updates
  - Changes frequently as users connect and disconnect

- Group Mapping
  - Answers: Which directory groups contain this user?
  - Retrieved from LDAP or Cloud Identity Engine
  - Allows Security policy to reference business roles

Common User Mapping Sources

- GlobalProtect
- Windows-based User-ID agent
  - Windows security event logs
  - WMI Probing
- PAN-OS integrated User-ID agent
  - Windows security event logs
  - WMI Probing
- RADIUS servers
- Authentication policy
- Terminal Server agent
- User-ID redistribution
- 3rd party identity logs can be forward to the firewall via Syslog or API



digitalscepter

GlobalProtect as a Preferred Mapping Source

- Directly associates an authenticated user with the assigned endpoint IP
- Supports remote and internally connected users
  - Internal gateways can provide User-ID without tunneling internal traffic
- Updates mappings as users change networks
- Reduces dependence on domain controller event monitoring
- Works well for laptops moving between wired, wireless and remote networks
- Provides a strong relationship between the authenticated user and endpoint

Windows Agent vs Integrated Agent

- Windows-Based UID Agent
  - Runs on a supported Windows Server
  - Monitors domain controllers and other authentication servers
  - Can provide mappings to multiple firewalls
  - Separates User-ID processing from the firewall
  - Often better suited for larger environments
- PAN-OS Integrated Agent
  - Runs directly on the firewall
  - Avoids deploying a separate Windows server
  - May be simpler for small environments
  - Causes each firewall to monitor authentication sources independently
  - Adds User-ID processing to the firewall management plane
  - Requires more permissions for Active Directory service account

User-ID Service Account Best Practices

- Use a dedicated account specifically for User-ID
- Grant only the permissions required by the selected monitoring method
- **Do not use a domain administrator account**
- Deny interactive login and do not grant the service account administrative access (including the UID Agent server)
- Use a long managed password
- Monitor the account for unexpected activity
- Document every firewall and agent using the account
- Rotate credentials through a controlled process
    - Did someone say CyberArk?

Syslog, XML API, and Authentication Policy

- Both the UID agent and the PAN-OS device can receive syslog messages and extrapolate IP-to-User mappings from them
  - This is particularly useful for wireless controllers and NAC systems
- Some 3rd party tools can integrate directly into the XML API which offers the benefit of a “logout” mechanism for clearing user mappings–by default they will age out after a configurable duration
- Authentication Policy can be used to trigger a Captive Portal to identify users. It may also be leveraged to force users to complete an MFA challenge in order to access desired resources.
  - A fun test is to force MFA on your favorite peers with a 1 minute timeout

Terminal Servers

- Multiple users may share one IP on Citrix or Windows RDS servers
- Standard IP-to-user mapping cannot distinguish concurrent users
- The Terminal Server agent assigns source-port ranges to individual users to track sessions from multiple users from a single IP
- The firewall maps the source IP and port range to the username
- Policy can then distinguish users sharing the same server
- Non-Windows shared systems may require XML API integration
- Port allocation should be tested before production deployment

Mapping Precedence and Timeouts

- A firewall may receive mappings for the same IP from multiple sources
- Conflicting sources can cause usernames to change unexpectedly
- Mapping methods should be intentionally selected
- Disable unreliable or duplicate sources
- Long timeouts can leave stale mappings
- Short timeouts can remove valid users too quickly
- Logout events should remove mappings before timeout where possible
- Shared computers and frequently reassigned addresses require shorter lifetimes

Group Mapping Best Practices

- In policy, leverage directory groups instead of individual users as much as possible
- Use secure LDAP or Cloud Identity Engine
- Configure redundant directory servers
- Use a minimally privileged bind account–it just needs to read group members, domain user is fine
- Limit group mapping to groups required by policy
- Avoid importing every directory group
- Use groups based on business roles
- Verify nested group behavior
- Document the directory owner for each policy group

Username Normalization

- The same user may appear as
  - DOMAIN\username
  - username@domain.example
  - Username
  - Email address
  - Distinguished name
- Best Practice:
  - Select a consistent username format
  - Ensure user mapping and group mapping resolve the same identity
  - Avoid mixing NetBIOS and DNS domain formats
  - Check for duplicate usernames across domains
  - Test users from every authentication source
  - Verify the format before creating Security policy

Multi-Domain Environments

- Define how usernames from each domain will be represented
- Use fully qualified usernames where ambiguity exists
- Configure directory connectivity to every required domain or forest
- Verify trust relationships and search scope
- Account for duplicate usernames
- Test nested groups across domain boundaries
- Confirm that every mapping source produces a format group mapping can resolve
- Limit directory searches to relevant domains and groups

User-ID Redistribution

- Shares mappings between firewalls and Panorama
- Eliminates requirement for every firewall to monitor every authentication source
- Useful when authentication occurs at one location and enforcement occurs elsewhere
- Supports centralized and large-scale User-ID designs
- Reduces repeated polling of domain controllers
- Requires redundancy when policy depends on redistributed mappings
- Distribution paths should be secured
- Avoid loops and duplicate redistribution paths

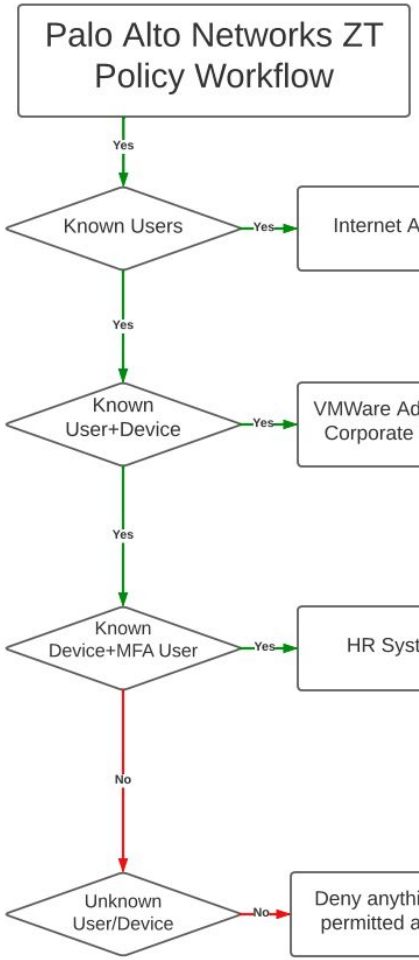
Enabling User-ID Safely

- User-ID should be enabled on zones containing source users
  - Enable it only where identity is needed
- Do not enable User-ID on untrusted, internet-facing zones
- Ensure the firewall sees the original client address
- Account for VPN, wireless and virtual desktop users
- Verify zone configuration before troubleshooting policy

Building User-Based Security Policy

- Combine identity with App-ID, destination and URL category
- Build rules around job role and business need
- Avoid broad rules that allow a group to access any application
- Place specific identity rules above general network rules
- Apply Security profiles to allowed traffic
- Include an explicit strategy for unknown users
- Validate users belonging to multiple groups

Zero Trust Policy Flow



Session has been identified with valid credentials

NAME	TAGS	TYPE	Source ZONE	Source ADDRESS	Source USER	Source DEVICE	Destination ZONE	Destination ADDRESS	Destination DEVICE	APPLICATI...	SERVICE	ACTION	PROFILE	OPTIONS
Known Users Internet Access	none	universal	inside	any	known-user	any	outside	any	any	any	application-default	Allow		

Valid Credentials and known device

NAME	TAGS	TYPE	Source ZONE	Source ADDRESS	Source USER	Source DEVICE	Destination ZONE	Destination ADDRESS	Destination DEVICE	APPLICATI...	SERVICE	ACTION	PROFILE	OPTIONS
Allow VMware Admins to vCenter	none	universal	inside	any	company\vmware_admins	corporate_secured	vmware	any	any	any	any	Allow		

Valid Credentials and known device

NAME	TAGS	TYPE	Source ZONE	Source ADDRESS	Source USER	Source DEVICE	Destination ZONE	Destination ADDRESS	Destination DEVICE	APPLICATI...	SERVICE	ACTION	PROFILE	OPTIONS
Allow HR Staff to HR Systems	none	universal	inside	any	company\hr_staff	corporate_secured	hr	hr_sy...	any	any	any	Allow		

NAME	TAGS	Source ZONE	Source ADDRESS	Source USER	Source DEVICE	Destination ZONE	Destination ADDRESS	Destination DEVICE	SERVICE	AUTHENTICATION ENFORCEMENT
Force MFA for HR Systems	none	inside	any	company\hr_staff	any	hr	hr_systems	any	any	force_mfra

MFA Enforced for Access

NAME	TAGS	TYPE	Source ZONE	Source ADDRESS	Source USER	Source DEVICE	Destination ZONE	Destination ADDRESS	Destination DEVICE	APPLICATI...	SERVICE	ACTION	PROFILE	OPTIONS
Deny All Unknown	none	universal	inside	any	any	any	any	any	any	any	any	Deny	none	

digitalscepter

App-ID Best Practices

What is App-ID

- Identifies applications independently of port, protocol or evasive techniques
- Continuously evaluates traffic as the session develops
- Provides application visibility without requiring separate App-ID configuration
- Allows Security policy to control the actual application rather than only a TCP or UDP port
- Supplies application context to logging, reporting, QoS and security inspection

What’s Wrong With Port-Based Policy

- Many applications share TCP 80 and TCP 443
- Applications can use nonstandard ports
- Malware can tunnel across commonly allowed services
- Opening a port permits every application capable of using that port
- Port-based rules provide limited visibility into application behavior
- Maintaining port-to-application mappings creates administrative overhead
- A rule permitting TCP 443 does not mean the traffic is legitimate web browsing

How App-ID Identifies Traffic

- App-ID uses multiple identification techniques
  - Application signatures
  - Protocol decoders
  - Behavioral and heuristic analysis
  - TLS and certificate information
- Identification may become more specific as additional packets are inspected

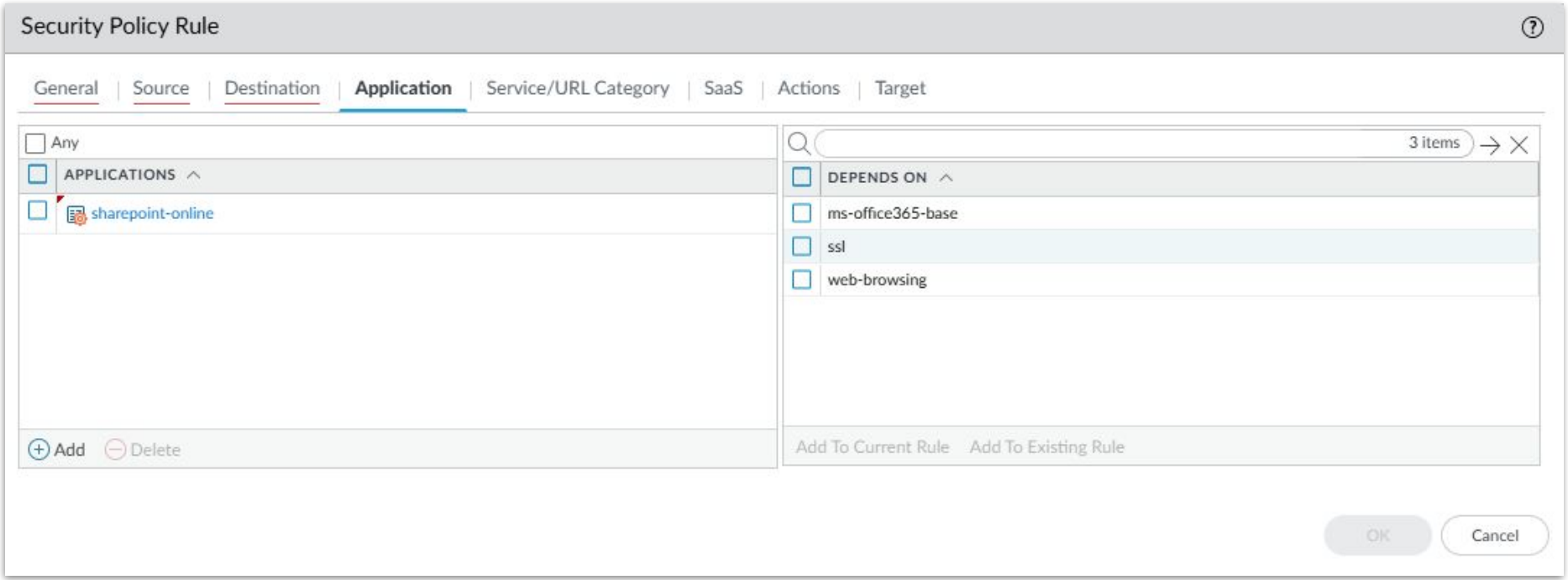
Applications and Services (Ports)

- Palo Alto allows controlling ports via services
- A given security policy with an application can be allowed to run under any service, a specific defined service (like TCP 443), or on the application-default ports
- Application-default restricts an application to it's known default ports which is maintained by Palo Alto
- The firewall accounts for encrypted and unencrypted application variants
- Should be used on the majority of application based allow rules, unless there is a desire to only permit an application on a specific port and the default supports multiple
- Application based deny rules should **always** have service set to "any"

Application Dependencies

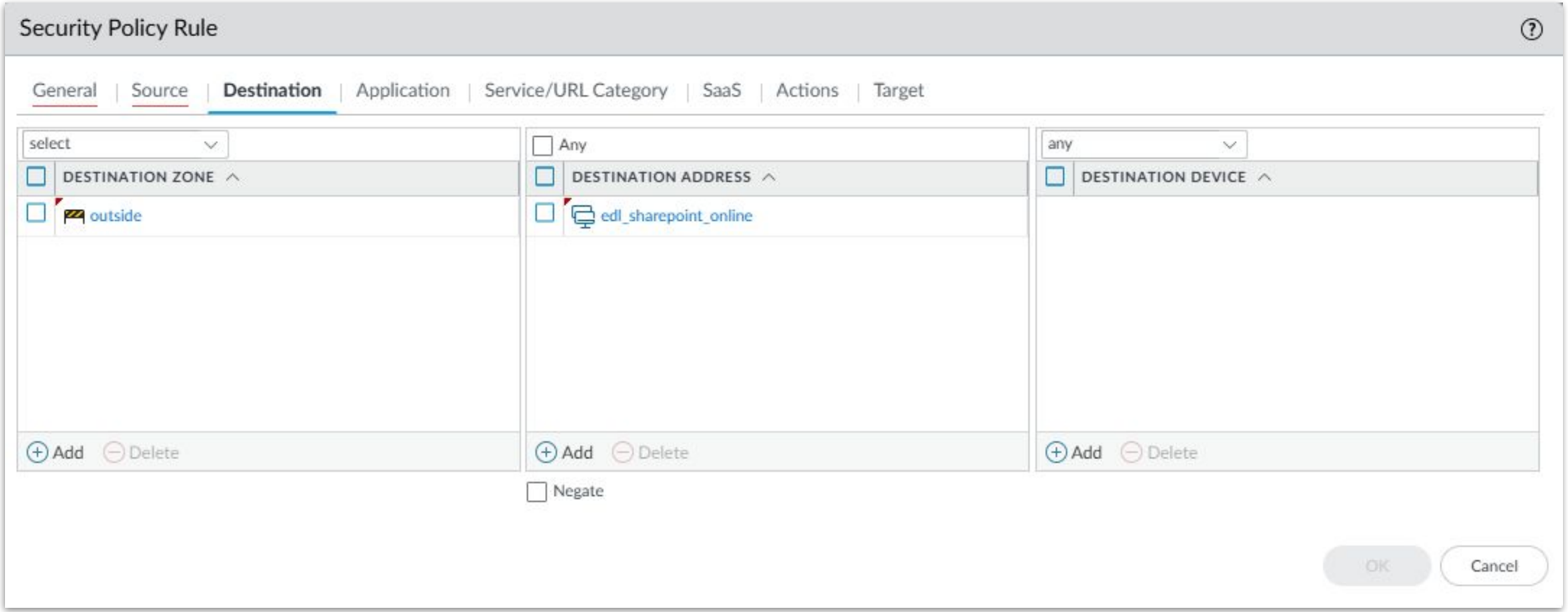
- Some applications require other applications to establish a connection
- Dependencies may include foundational services such as DNS, SSL or web browsing
- Missing dependencies can cause application failures
- PAN-OS identifies dependencies while building policy and during commit validation
- Dependencies should be reviewed rather than added blindly
- Parent and dependency applications may require different access scopes
- Security profiles should still be applied to dependency traffic

Application Dependencies



digitalscepter

Application Dependencies



digitalscepter

Application Groups

- Combine specific applications into a reusable static object
- Useful when several applications support one business function
- Simplify repeated application selections across rules

NAME	LOCATION	MEMBERS	APPLICATIONS
AG_ACTIVE_DIRECTORY	Shared	10	dns
			ms-netlogon
			ms-ds-smb
			ldap
			msrpc
			kerberos
			active-directory
			more...

digitalscepter

Application Filters

- Dynamically select applications based on characteristics
- Can filter by category, subcategory, technology, risk, or application tags
- Automatically include applications matching the selected criteria
- Reduce manual maintenance as new App-IDs are released
- Useful for broad block rules or controlled application categories
- Require careful review because future applications may automatically match

NAME	LOCATION	CATEGORY	SUBCATEGORY	TECHNOLOGY	RISK
af_proxies	Shared		proxy		
af_games	Shared		gaming		
af_p2p	Shared		file-sharing	peer-to-peer	
af_social media	Shared	collaboration	social-networking		1 2 3 4

Groups vs Filters

- Application Groups
  - Static membership
  - Administratively selected applications
  - Predictable and tightly controlled
  - Best for specific approved business applications

- Application Filters
  - Dynamic membership
  - Based on application characteristics or tags
  - Automatically adapts to newly released App-IDs
  - Best for application categories and broad control objectives

Groups vs Filters

- For inbound rules, a static application list is usually best

NAME	TAGS	Source				Destination			APPLICATI...	SERVICE	URL CATEGORY	ACTION	PROFILE	OPTIONS
		ZONE	ADDRESS	USER	DEVICE	ZONE	ADDRESS	DEVICE						
allow untrust to ext ftp server	none	untrust	any	any	any	dmz	extftp01_public	any	ftp	application-default	any	Allow		

- For general internet access policies leverage application filters

NAME	TAGS	Source				Destination			APPLICATION	SERVICE	URL CATEGORY	ACTION	PROFILE	OPTIONS
		ZONE	ADDRESS	USER	DEVICE	ZONE	ADDRESS	DEVICE						
permit internet	none	trust	any	any	any	untrust	any	any	af_business tools af_collaboration af_general internet	tcp_80 tcp_443	any	Allow		

- You may need specific policies that don't fit into app filters, use static apps for those

NAME	TAGS	Source				Destination			APPLICATI...	SERVICE	URL CATEGORY	ACTION	PROFILE	OPTIONS
		ZONE	ADDRESS	USER	DEVICE	ZONE	ADDRESS	DEVICE						
allow developers to github	none	trust	any	ds\developers	any	untrust	any	any	git-base github	tcp_80 tcp_443	github	Allow		

Positive Enforcement Model

- Define which applications users are permitted to access
- Allow only required applications
- Apply Security Profile Groups to permitted traffic
- Allow all other applications to fall through to default deny
- Prefer specific allow rules over attempting to block every unwanted application
- Review broad internet rules regularly
- Align application access with users, destinations and business functions

Building an App-ID Rule

- A well-designed application rule should answer
  - Who requires access
  - Which application is required
  - Where the user is connecting
  - Which destination provides the service
  - Whether the application is allowed only on default ports
  - Which Security Profile Group should apply
  - Who owns the business requirement
  - How the rule will be monitored and reviewed

		Source				Destination								
NAME	TAGS	ZONE	ADDRESS	USER	DEVICE	ZONE	ADDRESS	DEVICE	APPLICATI...	SERVICE	URL CATEGORY	ACTION	PROFILE	OPTIONS
allow developers to github	none	trust	any	ds\developers	any	untrust	any	any	git-base github	tcp_80 tcp_443	github	Allow		

digitalscepter

Avoid Broad Foundational Rules

- Examples of risky rules
  - Allow ssl and web-browsing to any destination
  - Allow application “any” over TCP 443
  - Allow all users to all SaaS applications
  - Allow required applications without destination restrictions
  - Place a general internet rule above specific App-ID rules (rule shadowing)
- Better approach
  - Use URL categories to constrain general web traffic
  - Create dedicated rules for sanctioned applications
  - Restrict applications by user group
  - Apply decryption where permitted
  - Review applications seen through broad fallback rules

App-ID and SSL Decryption

- App-ID can identify some encrypted applications using metadata and behavioral patterns
- Decryption exposes the payload and improves identification accuracy
- Many applications may otherwise remain classified as ssl
- Decryption enables more granular SaaS and web application identification
- App-ID policy and decryption policy should be designed together

Unknown Applications

- Unknown traffic appears as
  - Unknown-tcp
  - Unknown-udp
  - Unknown-p2p
- Possible causes
  - A custom internal application
  - A newly released commercial application
  - An unsupported protocol
  - Asymmetric routing
  - Insufficient traffic for classification
  - Evasive or malicious traffic

Unknown Applications

- Unknown traffic should be investigated rather than permanently allowed without review
- Review destinations and other session details to try to determine if unknown traffic is legitimate
- Create custom applications to identify previously unknown traffic or work with Palo Alto if it’s a commercial application

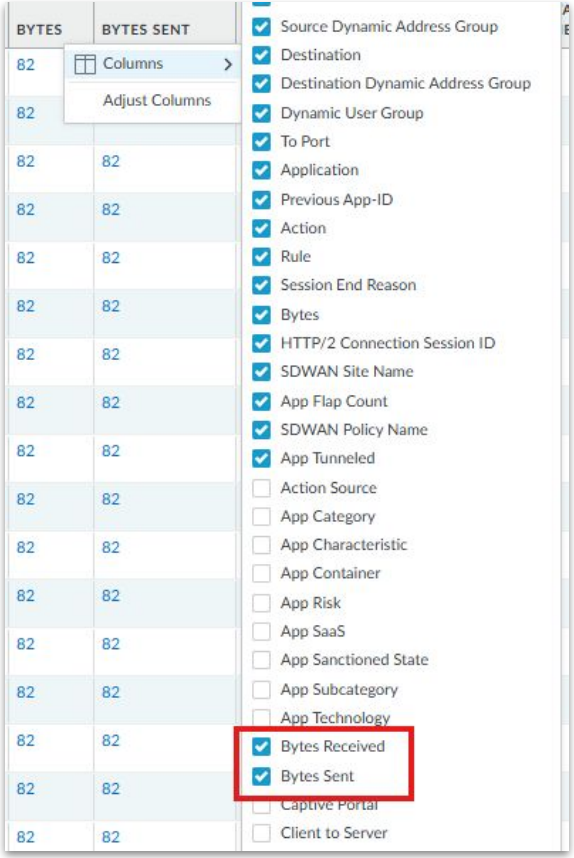
Incomplete and Insufficient-Data

- Incomplete
  - TCP handshake did not complete
  - Server did not respond
  - Routing or policy may be incorrect
  - Traffic may be scanning or probing
- Insufficient-Data
  - Session was established but too little payload was exchanged
  - Application closed before classification completed
  - Common with short-lived health checks or probes
  - App-ID did not receive enough data to identify an application

Incomplete Traffic

- Add the bytes received and bytes sent column in your traffic log view
- Bytes received = 0 means the firewall never received a reply (some caveats, but generally this is the case)

TO PORT	APPLICATION	PREVIOUS APP-ID	ACTION	RULE	SESSION END REASON	BYTES	BYTES SENT	BYTES RECEIVED
179	incomplete		allow	intrazone-default	aged-out	312	312	0
179	incomplete		allow	intrazone-default	aged-out	312	312	0
179	incomplete		allow	intrazone-default	aged-out	312	312	0
179	incomplete		allow	intrazone-default	aged-out	312	312	0
179	incomplete		allow	intrazone-default	aged-out	312	312	0



digitalscepter

Managing Custom Applications

- Create a custom App-ID when
  - An internal application is consistently classified as unknown
  - A proprietary protocol requires dedicated policy
  - A commercial application lacks a suitable App-ID
  - A specific application must be distinguished from other traffic on the same port
- Best practices
  - Use the narrowest reliable signature
  - Avoid signatures matching generic strings
  - Define appropriate parent applications and dependencies
  - Test against representative traffic
  - Monitor for false positives
  - Submit commercial unknown applications to Palo Alto Networks where appropriate

Policy Optimizer

- Policy Optimizer identifies
  - Port-based rules using application any
  - Applications observed on those rules
  - App-ID rules containing unused applications (overprovisioned)
  - Rule hit counts and usage information
- Benefits
  - Provides an evidence-based migration workflow
  - Helps prioritize risky or heavily used rules
  - Reduces manual traffic-log analysis
  - Supports ongoing rulebase maintenance

Migrating Port-Based Rules

- Recommended process
  a. Identify rules with application “any”
  b. Review applications observed over a representative period
  c. Confirm the applications with the business owner
  d. Clone or create a specific App-ID rule above the original
  e. Configure service application-default (or use services from original rule)
  f. Monitor the original rule for remaining traffic
  g. Investigate applications still hitting the original rule
  h. Disable and eventually remove the legacy rule

Migrating Port-Based Rules

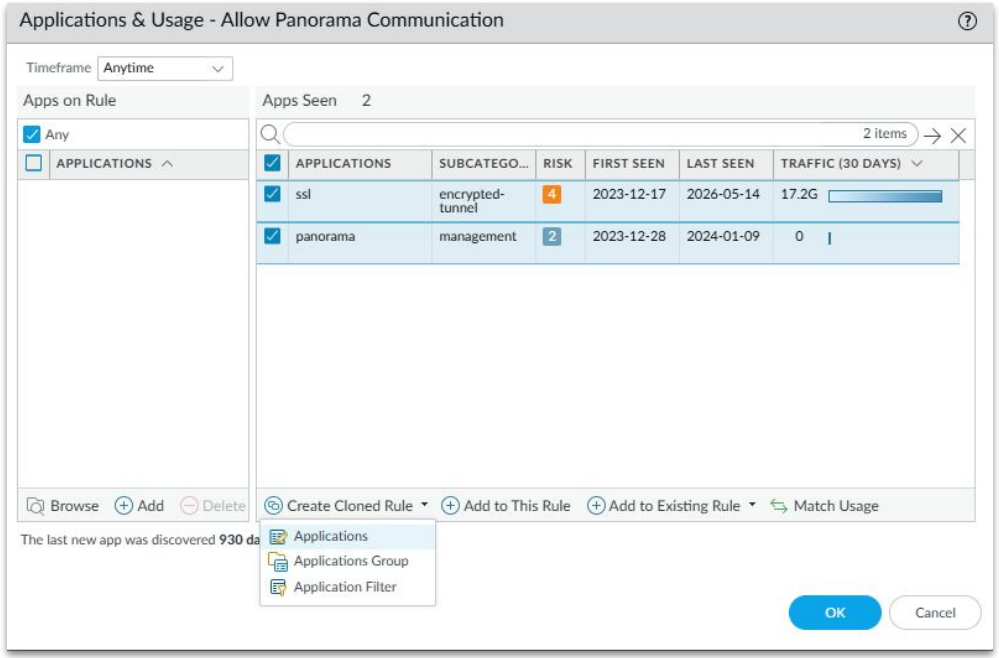
- Click on the number in “Apps Seen” column

			Source				Destination								Rule Usage	
	NAME	TAGS	ZONE	ADDRESS	USER	DEVICE	ZONE	ADDRESS	DEVICE	APPLICATION	SERVICE	ACTION	PROFILE	OPTIONS	TARGET	RULE USAGE	APPS SEEN
25	Allow Panorama Co...	internal	any	LAB_FIREWALLS PRD_FIREWALLS prdlpra01	any	any	any	LAB_FIREWAL... PRD_FIREWA... prdlpra01	any	any	any	Allow			any	Partially Used	2

digitalscepter

Migrating Port-Based Rules

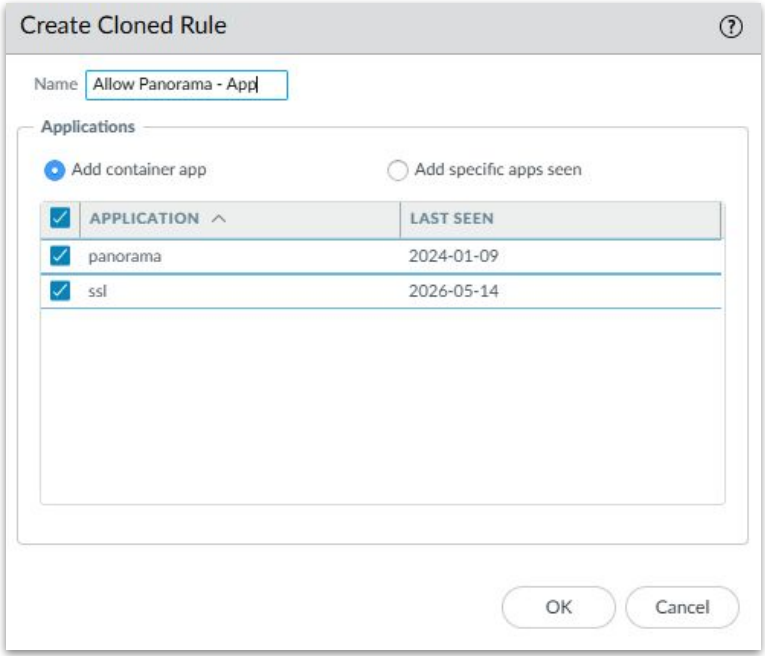
- Click **Create Cloned Rule** > **Applications**



digitalscepter

Migrating Port-Based Rules

- Provide a unique name, select desired applications, and click **OK**
- Add container app will select the parent application when a child application was detected
- In the example below, ms-ds-smb would be the container app for ms-ds-smbv2



digitalscepter

Q&A

digitalscepter

Thanks for attending

- Digital Scepter can be reached at digitalscepter.com or via email at sales@digitalscepter.com

digitalscepter

sales@digitalscepter.com
(888) 299-3718

digitalscepter.com