Active Defense Syslog

Introduce Active Defense Syslog

This Active Defense application is an Internet traffic forced syslog system. It was built based on the Palo Alto firewall and it mainly looks at traffic coming from the internet and finds out any IP / port scanning and vulnerability attack attempt activities and instructs the firewall to perform an explicit block action.

An example of how policy configuration makes use of this Active Defense system. The top policy to block IP addresses that match the Blacklist address object which is provided by Active Defense system. The bottom policy will feed all un-matched / implicit blocked traffic, plus all other public facing policies' logs to the system for data analysing where a Blacklist-IP will be produced.

				Source		Destination				
	NAME	TAGS	TYPE	ZONE	ADDRESS	ZONE	ADDRESS	APPLICATI...	SERVICE	ACTION
1	Active Defense Deny	none	universal	UNTRUST	AtiveDefense Blacklist	any	any	any	any	Reset Both
13	Active Defense sensor	none	universal	UNTRUST	any	UNTRUST	any	any	any	Reset Both

An example of list of bad guys. And the detail of why they got blacklisted. Firewall will block these public IPs by rule 1 above.

Blacklist Records

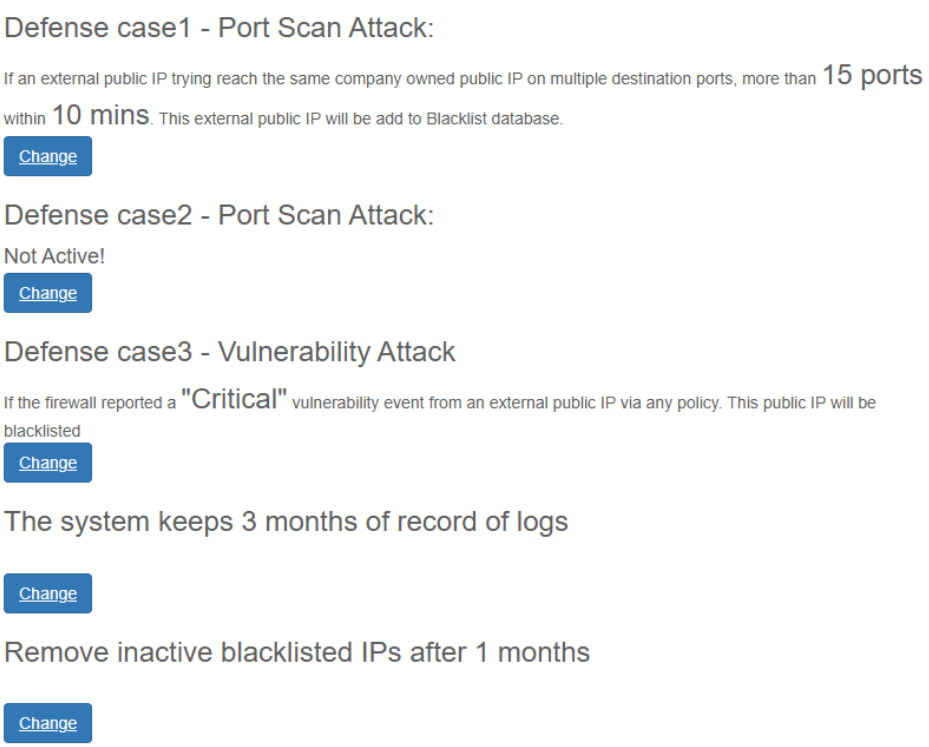
Type	Blacklisted IP
Vulnerability Scan v1	184.105.247.252
Port Scan Attack v1	45.142.193.118
Port Scan Attack v1	173.234.107.200
Port Scan Attack v1	51.161.172.223
Port Scan Attack v1	185.91.127.81
Port Scan Attack v1	165.154.205.78
Port Scan Attack v1	203.50.229.44

Syslog Records

Logfile: LAB-FW-01
Hostname: LAB-FW-01
Source Country: United States
Source IP: 184.105.247.252 Blacklist it
Source Port: 41890
User Account:
Destination Country: Australia
Destination IP:
Destination Port: 443
Threat Name: Palo Alto Networks GlobalProtect OS Command Injection Vulnerability(95187)
Threat Type: vulnerability
Severity: critical
Rule Name: GlobalProtect_Portal
Action: reset-both
Log Type: THREAT
Last Seen: 2025/02/20 11:49:43

		RECEIVE TIME	TYPE	FROM ZONE	SOURCE	TO ZONE	TO PORT	APPLICATION	ACTION	RULE	SESSION END REASON	SOURCE COUNTRY
		02/20 12:29:35	drop	UNTRUST	203.50.229.44	UNTRUST	0	ping	drop	Active Defense Deny	policy-deny	Australia
		02/20 12:29:33	drop	UNTRUST	165.154.205.78	UNTRUST	8060	not-applicable	reset-both	Active Defense Deny	policy-deny	Singapore
		02/20 12:29:26	drop	UNTRUST	51.161.172.223	UNTRUST	4821	not-applicable	reset-both	Active Defense Deny	policy-deny	Canada
		02/20 12:29:21	drop	UNTRUST	173.234.107.200	UNTRUST	16066	not-applicable	reset-both	Active Defense Deny	policy-deny	Australia
		02/20 12:29:10	drop	UNTRUST	173.234.107.200	UNTRUST	16066	not-applicable	reset-both	Active Defense Deny	policy-deny	Australia
		02/20 12:28:35	drop	UNTRUST	203.50.229.44	UNTRUST	0	ping	drop	Active Defense Deny	policy-deny	Australia
		02/20 12:28:20	drop	UNTRUST	165.154.205.78	UNTRUST	1027	not-applicable	reset-both	Active Defense Deny	policy-deny	Singapore
		02/20 12:28:04	drop	UNTRUST	45.142.193.118	UNTRUST	39940	not-applicable	reset-both	Active Defense Deny	policy-deny	Romania
		02/20 12:27:45	drop	UNTRUST	173.234.107.200	UNTRUST	16066	not-applicable	reset-both	Active Defense Deny	policy-deny	Australia
		02/20 12:27:35	drop	UNTRUST	203.50.229.44	UNTRUST	0	ping	drop	Active Defense Deny	policy-deny	Australia
		02/20 12:27:30	drop	UNTRUST	165.154.205.78	UNTRUST	8600	not-applicable	reset-both	Active Defense Deny	policy-deny	Singapore
		02/20 12:27:21	drop	UNTRUST	45.142.193.118	UNTRUST	39905	not-applicable	reset-both	Active Defense Deny	policy-deny	Romania
		02/20 12:27:12	drop	UNTRUST	51.161.172.223	UNTRUST	4821	not-applicable	reset-both	Active Defense Deny	policy-deny	Canada
		02/20 12:26:52	drop	UNTRUST	173.234.107.200	UNTRUST	16066	not-applicable	reset-both	Active Defense Deny	policy-deny	Australia
		02/20 12:26:48	drop	UNTRUST	185.91.127.81	UNTRUST	12983	not-applicable	reset-both	Active Defense Deny	policy-deny	Germany
		02/20 12:26:42	drop	UNTRUST	51.161.172.223	UNTRUST	4821	not-applicable	reset-both	Active Defense Deny	policy-deny	Canada

Finally, users can tune the settings of how the Defense engine runs. Such as the number of ports per certain mins to be defined as port scan activity. Blacklist a public IP that is triggered by Critical-level or High-level vulnerability attempts. Keeping firewall logs for a number of months and remove blacklist IP after number of months without being detected with any bad activity.



Install Active Defense Syslog System

Assume you have an Ubuntu server installed and enabled SSH with full access

- Ubuntu server. 24.04.02

To make sure your server can utilize all disk space

sudo lvextend -l +100%FREE /dev/ubuntu-vg/ubuntu-lv
sudo resize2fs /dev/mapper/ubuntu--vg-ubuntu--lv

Rsyslog application which is available out-of-box with Ubuntu server.

- Edit file *"sudo nano /etc/rsyslog.conf"*
    - Find and un-hash below two line:
      module(load="imudp")

      input(type="imudp" port="514")
    - Add below 3 lines to the end of the file

      $template firewall,"/var/log/Firewall/%HOSTNAME%.log"

      if $fromhost != 'Whatever-the-server-hostname-is' then ?firewall

      & stop
- Create a folder for storing logs and give full access permission to all users

  sudo mkdir /var/log/Firewall

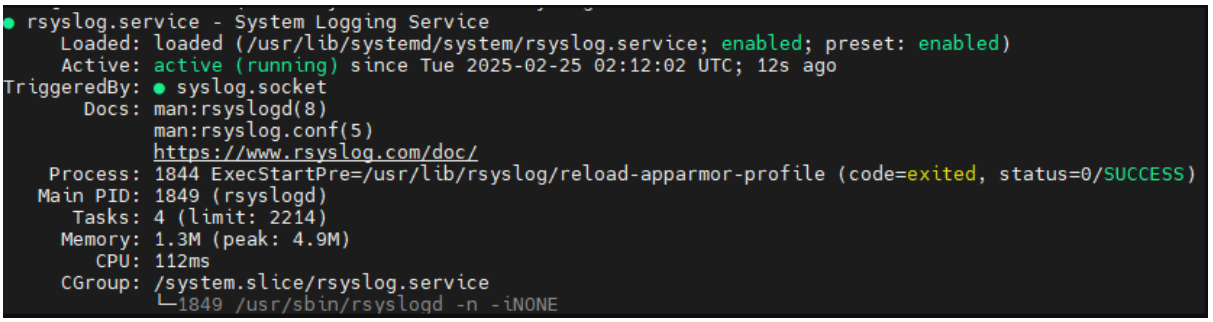
  sudo chmod 777 /var/log/Firewall
- Give the application the ability to restart the Rsyslog service
    - Edit file "sudo nano /etc/sudoers" add below line to the end of the file

ALL ALL=(ALL) NOPASSWD: /usr/bin/systemctl restart rsyslog.service

- Restart and verify Rsyslog service.

  sudo systemctl restart rsyslog.service

  systemctl status rsyslog.service



Install Django Framework

sudo apt-get update

sudo apt-get install python3-django

sudo apt-get install python3-pip python3-venv

Create virtual environment and install Django

cd /

sudo mkdir Automation

sudo chmod 777 Automation

python3 -m venv Automation

cd Automation

source bin/activate

pip3 install Django

```
user@activedefense:/$ sudo mkdir Automation
user@activedefense:/$ sudo chmod 777 Automation
user@activedefense:/$ python3
python3            python3.12         python3.12-config  python3-config
user@activedefense:/$ python3 -m venv Automation
user@activedefense:/$ ls
Automation  bin.usr-is-merged  cdrom  etc   lib    lib.usr-is-merged  media  op
bin         boot               dev    home  lib64  lost+found         mnt    pr
user@activedefense:/$ cd Automation/
user@activedefense:/Automation$ ls
bin  include  lib  lib64  pyvenv.cfg
user@activedefense:/Automation$ source bin/activate
(Automation) user@activedefense:/Automation$ pip3 install django
Collecting django
  Downloading Django-5.1.6-py3-none-any.whl.metadata (4.2 kB)
Collecting asgiref<4,>=3.8.1 (from django)
  Downloading asgiref-3.8.1-py3-none-any.whl.metadata (9.3 kB)
Collecting sqlparse>=0.3.1 (from django)
  Downloading sqlparse-0.5.3-py3-none-any.whl.metadata (3.9 kB)
Downloading Django-5.1.6-py3-none-any.whl (8.3 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 8.3/8.3 MB 6.4 MB/s eta 0:00:00
Downloading asgiref-3.8.1-py3-none-any.whl (23 kB)
Downloading sqlparse-0.5.3-py3-none-any.whl (44 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 44.4/44.4 kB 3.7 MB/s eta 0:00:00
Installing collected packages: sqlparse, asgiref, django
Successfully installed asgiref-3.8.1 django-5.1.6 sqlparse-0.5.3
(Automation) user@activedefense:/Automation$
```

Create web applications and install dependent libraries

- Perform below commands under directory /Automation within the virtual environment

    django-admin startproject ActiveDefense .

    python3 manage.py startapp Login

    python3 manage.py startapp Syslog

    pip3 install pathlib netifaces datetime apscheduler sqlalchemy

    pip3 install python-dateutil gunicorn

Copy the source code files and install the system

We will use sftp to copy the source files into created directories from above

If you are using a Windows computer you can use MoxaXterm free application to do this. Linux users can perform this task natively using Terminal.

- Start from the directory where you have downloaded source files. SFTP to the server. Copy and override everything into the remote server with below command

    cd /<the directory on your local computer where you downloaded the source code>

    sftp user@<server IP>

    cd /Automation

    *put -R **

    exit

- Install / create PostgreSQL database engine with below command

  sudo apt install postgresql postgresql-contrib libpq-dev

  source /Automation/bin/activate

  pip3 install psycopg2-binary

  - create Database

    sudo -u postgres psql

    SQL commands below:

    CREATE DATABASE activedefense;

    CREATE USER activedefenseuser WITH PASSWORD 'activedefensepassword;

    ALTER ROLE activedefenseuser SET client_encoding TO 'utf8';

    ALTER ROLE activedefenseuser SET default_transaction_isolation TO 'read committed';

    ALTER ROLE activedefenseuser SET timezone TO 'Australia/Sydney';

    ALTER ROLE activedefenseuser WITH CREATEDB;

    GRANT ALL PRIVILEGES ON DATABASE activedefense TO activedefenseuser;

    ALTER DEFAULT PRIVILEGES IN SCHEMA public

    GRANT ALL PRIVILEGES ON TABLES TO activedefenseuser;

    \q

  - Temporary rename the app.py file to avoid error during the database initial setup

    cd /Automation/Syslog

    mv apps.py apps.py.tmp

    cd ..

  - create Database

    python3 manage.py makemigrations

```
(Automation) user@activedefense:/Automation$ python3 manage.py makemigrations
Migrations for 'Syslog':
  Syslog/migrations/0001_initial.py
    + Create model Blacklist
    + Create model DefenseSetting
    + Create model EmailSetting
    + Create model JobLock
    + Create model Logfile
    + Create model Recipient
    + Create model PATrafficLog
(Automation) user@activedefense:/Automation$ cd ActiveDefense/
```

    python3 manage.py migrate

```
(Automation) user@activedefense:/Automation$ python3 manage.py migrate
Operations to perform:
  Apply all migrations: Syslog, admin, auth, contenttypes, sessions
Running migrations:
  Applying Syslog.0001_initial... OK
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying sessions.0001_initial... OK
(Automation) user@activedefense:/Automation$
```

- Create superuser

  python3 manage.py createsuperuser

```
(Automation) user@activedefense:/Automation$ python3 manage.py createsuperuser
Username (leave blank to use 'user'): user
Email address:
Password:
Password (again):
Superuser created successfully.
(Automation) user@activedefense:/Automation$
```

- Rename the apps.py.tmp back to apps.py

  cd /Automation/Syslog/

  mv apps.py.tmp apps.py

  cd ..

Start the application

- Apply executable permissions to startup scripts

  *chmod +x /Automation/run.sh**

- Setup auto start @reboot

  sudo nano /etc/crontab

  - add below line to the bottom and save / exit

    @reboot user /bin/bash -c "/Automation/run.sh"

Setup and configure the web server

- Setup Nginx (Engine X) as front-end web server
    - Install Nginx

        sudo apt-get install nginx

    - Setup web site in nginx by creating file “django” in /etc/nginx/sites-available/

        sudo nano /etc/nginx/sites-available/django

        - Copy below to the file, save and exit

        server {

        listen 443 ssl;

        server_name <server FQDN or IP Address>;

        ssl_certificate /Automation/Cert/server.crt;

        ssl_certificate_key /Automation/Cert/server.key;

        location / {

        proxy_pass http://127.0.0.1:8000;

        proxy_set_header Host $host;

        proxy_set_header X-Real-IP $remote_addr;

        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;

        proxy_set_header X-Forwarded-Proto $scheme;

        }

        location /static/ {

        alias /Automation/static/;

        }

        }

        server {

        listen 80;

        server_name your.domain.or.ip;

        return 301 https://$host$request_uri;

        }

- Start Nginx
  sudo ln -s /etc/nginx/sites-available/django /etc/nginx/sites-enabled/
  sudo nginx -t
  sudo systemctl restart nginx

- Set the correct timezone before Rebooting the server.

  sudo timedatectl set-timezone <local timezone such as Australia/Sydney>

  sudo reboot now

- The system will startup automatically. Access the application using:

  https://server-ip-address

  username password has been created as superuser above

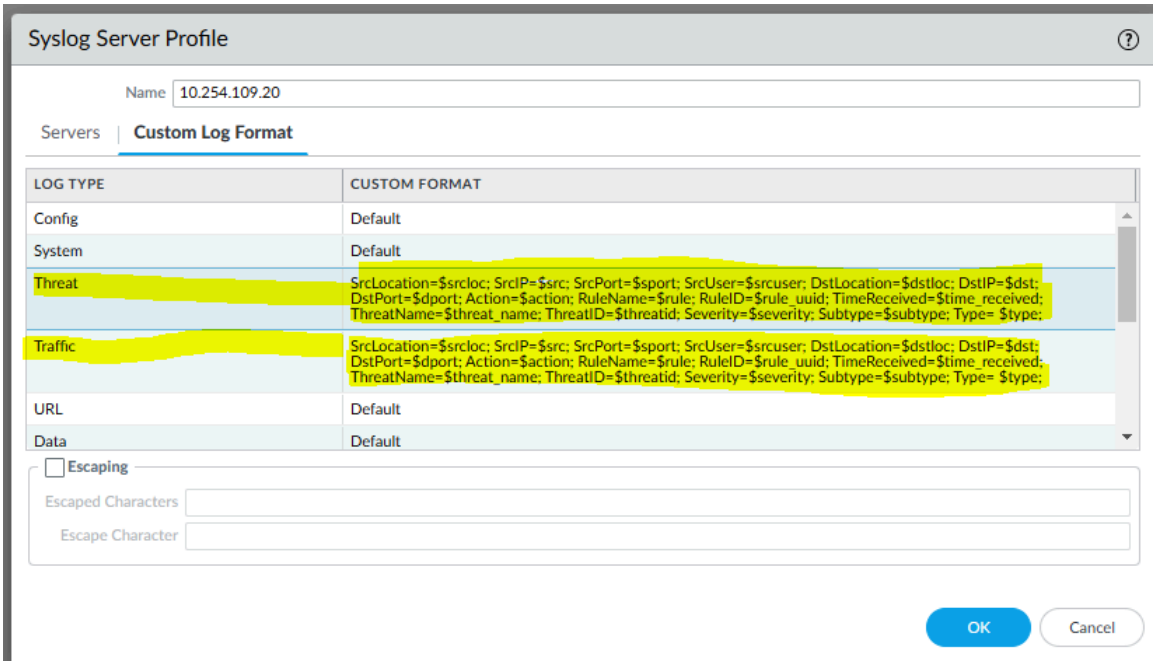
Setup Palo Firewall and forward the logs

Setup syslog at Palo firewall

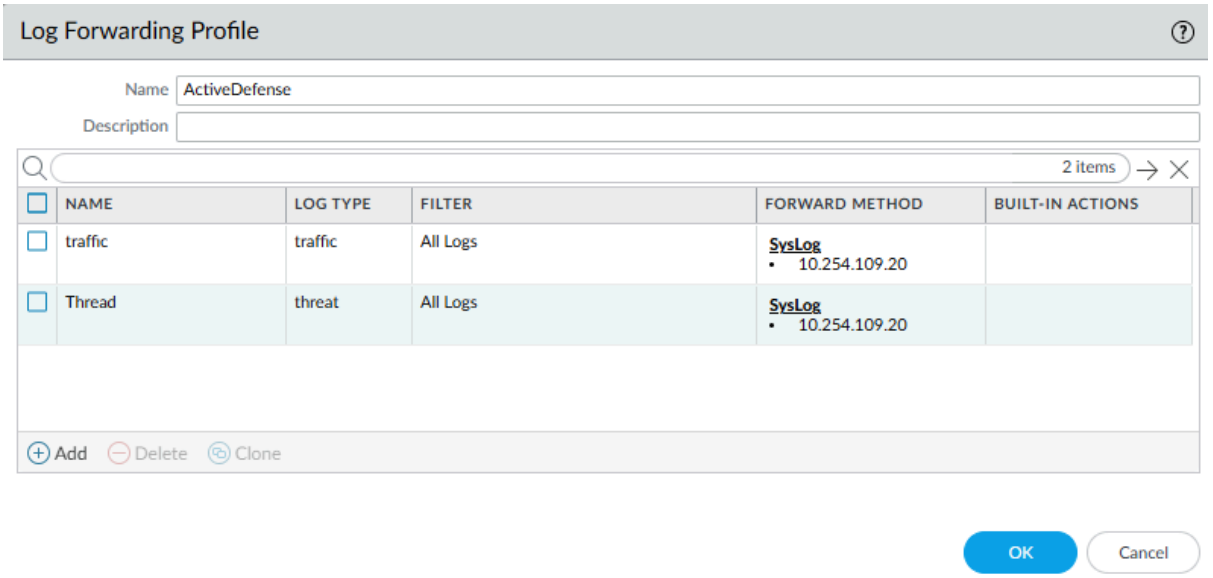
Device → Server Profiles → Syslog → Add new

Add the IP address of the Defense system as syslog server. Copy below log format string to both Thread and Traffic log type

SrcLocation=$srcloc; SrcIP=$src; SrcPort=$sport; SrcUser=$srcuser; DstLocation=$dstloc; DstIP=$dst; DstPort=$dport; Action=$action; RuleName=$rule; RuleID=$rule_uuid; TimeReceived=$time_received; ThreatName=$threat_name; ThreatID=$threatid; Severity=$severity; Subtype=$subtype; Type= $type;



Objects → Log Forwarding → Add new



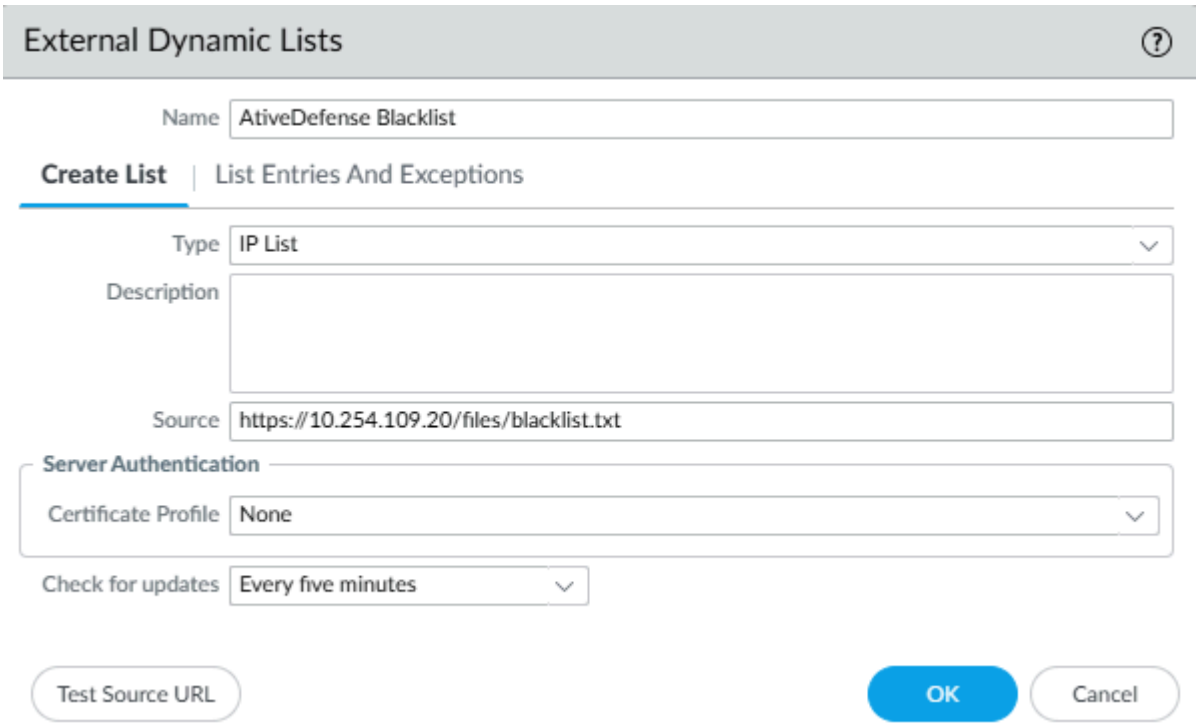
Create blacklist address object

Objects → External Dynamic Lists → Add new

Type: IP List

Source: https://<server-IP>/files/blacklist.txt

Check for update: Every 5 mins



Create the Active Defense policies

- The Fist policy needs to be at the top position

Source Zone: Internet/Untrust

Source: address: <ActiveDefense Blacklist created at above>

Destination Zone: any

Destination Address: any

Action: Deny

Log forward: <ActiveDefense Syslog>

Note: best practice is to forward the blocked logs. System will know if any known blacklisted IPs are still attacking you. System will reset its time within the block-window (default 1 month). Otherwise, system will remove it after block-window regardless. Eventually it will get block again as a new IP

NAME	TAGS	TYPE	ZONE	ADDRESS	ZONE	ADDRESS	APPLICATI...	SERVICE	ACTION	PROFILE	OPTIONS
Active Defense Deny	none	universal	UNTRUST	AtiveDefense Blacklist	any	any	any	any	Reset Both	none	

- The Second policy needs to be at the bottom position.

This policy needs to be at the bottom whenever a new policy is added in the future. Basically, it captures all un-matched traffic and forwards the logs to the syslog.

Source Zone: Internet / Untrust

Source Address: any

Destination Zone: any

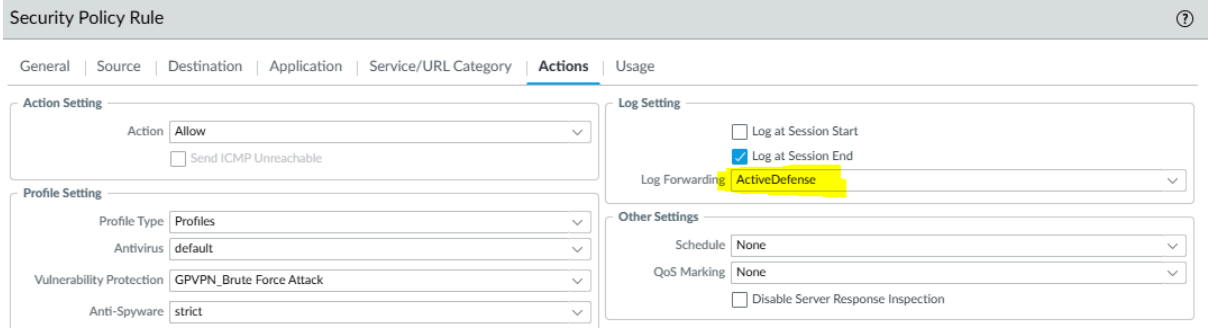
Action: Deny or reset

Log forward: <ActiveDefense Syslog>

Note: We need to forward all un-matched traffic that ActiveDefense can tell if anyone trying to do port scanning on you. If you don't have this catch-all rule. ActiveDefense can only block vulnerability detected IP.

NAME	TAGS	TYPE	ZONE	ADDRESS	ZONE	ADDRESS	APPLICATI...	SERVICE	ACTION	PROFILE	OPTIONS
Active Defense sensor	none	universal	UNTRUST	any	UNTRUST	any	any	any	Reset Both	none	

- Apply the ActiveDefense syslog to all other Internet facing policy such as the Globalprotect one as below



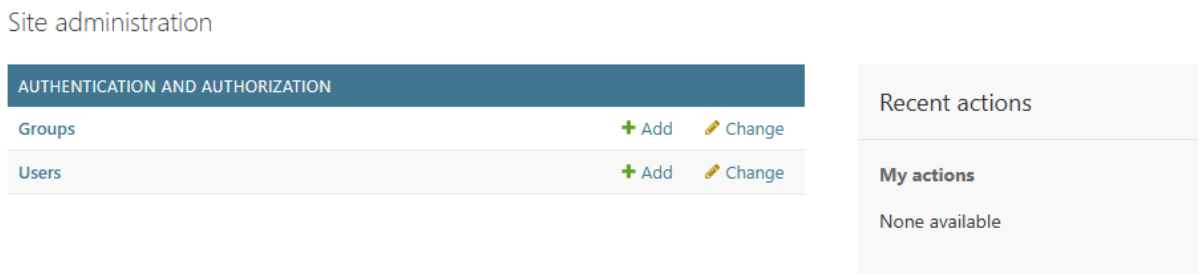
Note: ActiveDefense will find out if anyone tries to do vulnerability attacks on you and blacklist them. This type of policy usually has security Profile attached such as Vulnerability Protection etc.

Commit all above changes and the firewall is completed.

Setup Active Defense Syslog

Login to GUI and change password or create your own account

https://<server-ip>/admin



Setup SMTP server and recipients for notification. System will email recipients if new IP address added to blacklist (optional)

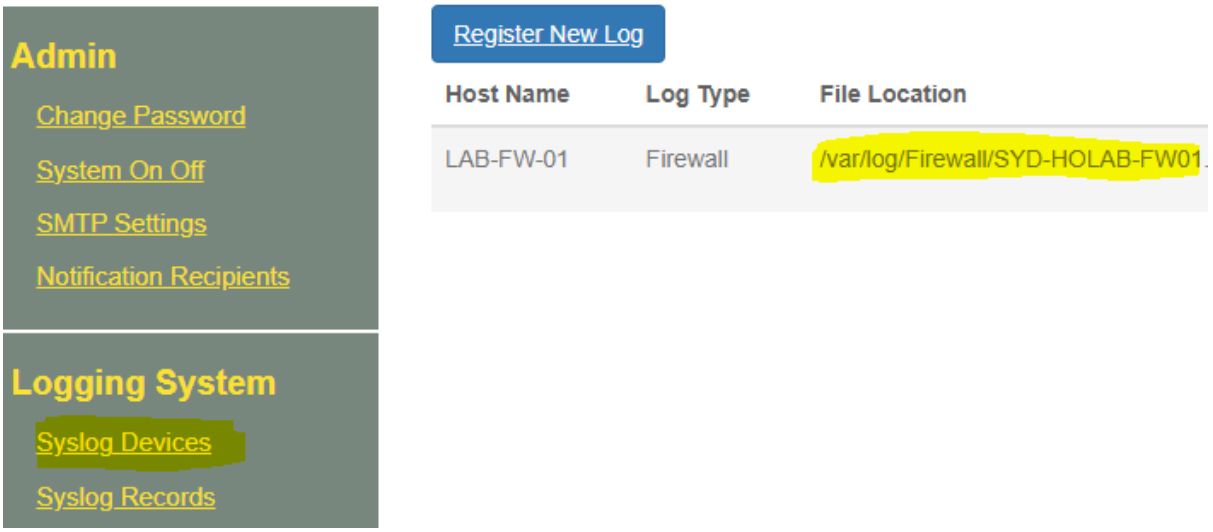
Find and add the syslog file location

New log data will be saved at /var/log/Firewall/ directory before converting into Database records. The log file name is the hostname of the firewall with extension ".log"

You can SSH to the server provided (above) and find out the log files.

Command: ls /var/log/Firewall/

Register the log file and path to the system



Tune the Active Defense settings to meet your need under mane "ActiveDefense Settings"

- Defense case 1 – port scanning (multiple ports)

Default setting. It is targeting a single public IP being tried with more than 10 ports in the last 10mins window. You can relax it by increasing the number of ports or lowering the time. Such as 20+ ports or in 8 mins. Be aware that the defence engineer runs every 5 mins. If the time is set to lower than 5 mins. It will skip logs

- Defense case 2 – Port scanning (multiple IPs)

  Default setting: system will find out if someone tries to find out open ports on all of your public IPs. Such as targeting https port (443) on all your public IPs. It is set to 10 of your public IP being tried on a single port in the last 10 mins. You can relax it by increasing the number of public IPs or reducing time. If you don't own multiple public IPs. You can disable this by setting the time to 0.

- Defense case 3 – Vulnerability detection

  By default. It blacklists any external public IP who had triggered "Critical" vulnerability alerts. You can make it more aggressive to set the value to 2. It will also blacklist someone who triggered a "High" vulnerability alert. Set to 0 to disable this function if you don't have Threat protection on your firewall.

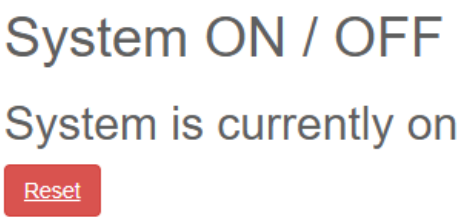
- The other two settings are Database keeping.

  By default, it is set to keep 3 month's log records and release the blacklisted IP if it has been doing bad things in a month.

Start the system and verify running

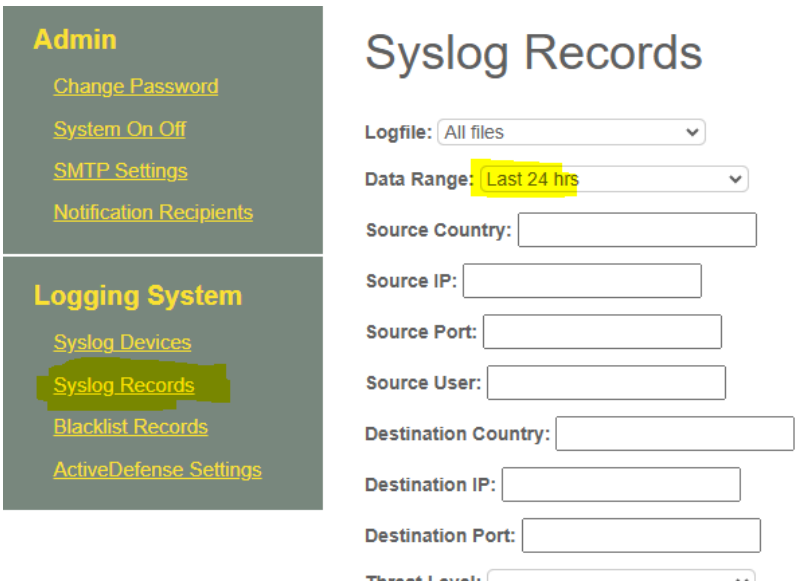
"System On Off" is to start or restart the scheduled jobs when needed. If you can see any new traffic logs in "Syslog Records". Do a "Reset" and "start". It will restart the defense engine. The logs will always be received even if the engine is not running. It will keep in /var/log/Firewall. the log file will grow until the engine runs again. Logs will convert into DB and the log file size gets reduced.

The engine will not auto start when you restart the server. You will need to login to the GUI and start the defense system manually or "reset / start" it every time you reboot the server.



To verify it's running. You can simply look at the last 24 hours logs. confirm by the time of the record

Or SSH to the server and look at the log file size. A .tmp file will be created when the system converts the logs into Database. Then the log file will get reduced and the .tmp file should be gone after the scheduled job is completed. It runs every 5 mins.



Limitation

If you have a large amount of logs and the server can not process it within 5 mins. Then you need a faster system to keep up with the log growing. Spread to multiple servers for multiple firewalls. Or external the scheduling time to every 10mins for example. It will require edit to the source code:

/Automation/Syslog/scheduler.py

```
def run():
    global scheduler
    if not job_exists('FletchLog_id01'):
        scheduler.add_job(FletchLog, 'interval', minutes = 5, max_instances = 1, misfire_grace_time=60, id = 'FletchLog_id01', replace_e
        time.sleep(30)
    if not job_exists('defense_id01'):
        scheduler.add_job(defense.run, 'interval', minutes = 5, max_instances = 1, misfire_grace_time=60, id = 'defense_id01', replace_e
    if not job_exists('remove_old_blacklist_id01'):
        scheduler.add_job(remove_old_blacklist, CronTrigger(hour = 0, minute = 0, timezone = 'Australia/Sydney'), max_instances = 1, mis
    if not job_exists('delete_old_logs_id01'):
        scheduler.add_job(delete_old_logs, CronTrigger(hour = 1, minute = 0, timezone = 'Australia/Sydney'), max_instances = 1, misfire_
    if not scheduler.running:
        scheduler.start()
    job_lock, created = JobLock.objects.get_or_create(job_name='System_ON_OFF')
    job_lock.is_running = True
    job_lock.save()
```

Note: the “sleep time” is the gap between two jobs. Database gets locked every action. Job gets delayed automatically when DB gets locked. Graceful period of 1 mins before the job gives up and waits for the next run.