#

##############################################################################

#

# This is the configuration file for psad (the Port Scan Attack Detector).

# Normally this file gets installed at /etc/psad/psad.conf, but can be put

# anywhere in the filesystem and then the path can be specified on the

# command line argument "-c <file>" to psad. All three psad daemons (psad,

# kmsgsd, and psadwatchd) reference this config file. Note that kmsgsd is

# generally deprecated since by default psad parses iptables log messages

# directly from the file where syslog writes them. Further, psadwatchd is

# not required if running on a Linux system that already has a process

# monitoring and restarting capability built-in such as provided by the

# upstart daemon.

#

# Each line has the form "<variable name> <value>;". Note the semi-

# colon after the <value>. All characters after the semicolon will be

# ignored to provide space for comments.

#

##############################################################################

#

### Supports multiple email addresses (as a comma separated

### list).

EMAIL\_ADDRESSES INBOX;

### Machine hostname

HOSTNAME CHANGEME;

### Specify the home and external networks. Note that by default the

### ENABLE\_INTF\_LOCAL\_NETS is enabled, so psad automatically detects

### all of the directly connected subnets and uses this information as

### the HOME\_NET variable.

HOME\_NET any;

EXTERNAL\_NET any;

### The FW\_SEARCH\_ALL variable controls how psad will parse iptables

### messages. If it is set to "Y" then psad will parse all iptables

### messages for evidence of scan activity. If it is set to "N" then

### psad will only parse those iptables messages that contain logging

### prefixes specified by the FW\_MSG\_SEARCH variable below. Logging

### prefixes are set with the --log-prefix command line option to iptables.

### Setting FW\_SEARCH\_ALL to "N" is useful for having psad only analyze

### iptables messages that are logged out of a specific iptables chain

### (multiple strings can be searched for, see the comment above the

### FW\_MSG\_SEARCH variable below) or a specific logging rule for example.

### FW\_SEARCH\_ALL is set to "Y" by default since usually people want psad

### to parse all iptables messages.

FW\_SEARCH\_ALL Y;

### The FW\_MSG\_SEARCH variable can be modified to look for logging messages

### that are specific to your firewall configuration (specified by the

### "--log-prefix" option. For example, if your firewall uses the

### string "Audit" for packets that have been blocked, then you could

### set FW\_MSG\_SEARCH to "Audit"; The default string to search for is

### "DROP". Both psad and kmsgsd reference this file. NOTE: You can

### specify this variable multiple times to have psad search for multiple

### strings. For example to have psad search for the strings "Audit" and

### "Reject", you would use the following two lines:

#FW\_MSG\_SEARCH Audit;

#FW\_MSG\_SEARCH REJECT;

FW\_MSG\_SEARCH DROP;

### Set the type of syslog daemon that is used. The SYSLOG\_DAEMON

### variable accepts four possible values: syslogd, syslog-ng, ulogd,

### or metalog. Note: this variable is only used if ENABLE\_SYSLOG\_FILE is

### disabled, and this in turn will mean that the legacy kmsgsd daemon will

### collect firewall logs from syslog via the old named pipe mechanism.

SYSLOG\_DAEMON syslogd;

### What type of interface configuration do you use? Set this variable to

### "iproute2" if you want to use the iproute2 type configuration.

### iproute2 does not use aliases for multi-homed interfaces and

### ifconfig does not show secondary addresses for multi-homed interfaces.

#IFCFGTYPE iproute2;

IFCFGTYPE ifconfig;

### Danger levels. These represent the total number of

### packets required for a scan to reach each danger level.

### A scan may also reach a danger level if the scan trips

### a signature or if the scanning ip is listed in

### auto\_ips so a danger level is automatically

### assigned.

DANGER\_LEVEL1 5; ### Number of packets.

DANGER\_LEVEL2 15;

DANGER\_LEVEL3 150;

DANGER\_LEVEL4 1500;

DANGER\_LEVEL5 10000;

### Set the interval (in seconds) psad will use to sleep before

### checking for new iptables log messages

CHECK\_INTERVAL 5;

### Search for snort "sid" values generated by fwsnort

### or snort2iptables

SNORT\_SID\_STR SID;

### For systems with an init daemon like 'upstart' that offer built-in process

### monitoring, it is not necessary to run the psadwatchd daemon. For such

### systems, the following variable can be set to 'N' to disable psadwatched

### altogether.

ENABLE\_PSADWATCHD N;

### Set the minimum range of ports that must be scanned before

### psad will send an alert. The default is 1 so that at

### least two port must be scanned (p2-p1 >= 1). This can be set

### to 0 if you want psad to be extra paranoid, or 30000 if not.

PORT\_RANGE\_SCAN\_THRESHOLD 1;

### For IP protocol scan detection (nmap -sO). While it may be relatively

### common for a host to trigger on tcp, udp, and icmp, it is more unusual if

### a host triggers on, say, five different IP protocols

PROTOCOL\_SCAN\_THRESHOLD 5;

### If "Y", means that scans will never timeout. This is useful

### for catching scans that take place over long periods of time

### where the attacker is trying to slip beneath the IDS thresholds.

ENABLE\_PERSISTENCE Y;

### This is used only if ENABLE\_PERSISTENCE = "N";

SCAN\_TIMEOUT 3600; ### seconds

### Specify how often to timeout old scan data relative to CHECK\_INTERVAL

### iterations. This feature is only used if ENABLE\_PERSISTENCE is disabled.

### Note that for psad processes that have tracked a lot of scans, it is

### advisable to leave this threshold at the default value of 5 or greater

### because the scan tracking hash may be quite large.

PERSISTENCE\_CTR\_THRESHOLD 5;

### Limit the number of src->dst IP pairs that psad will track. The default

### is zero (i.e. unlimited), but if psad is running on a system with limited

### memory, this can be handy to restrict psad's memory usage. It is best to

### combine this option with disabling ENABLE\_PERSISTENCE so that older scans

### are deleted and therefore newer scans will on average continue to be

### tracked. A good non-zero value is, say, 50000, but this will vary

### depending on available system memory.

MAX\_SCAN\_IP\_PAIRS 0;

### If "Y", means all signatures will be shown since

### the scan started instead of just the current ones.

SHOW\_ALL\_SIGNATURES N;

### Allow reporting methods to be enabled/restricted. This keyword can

### accept values of "nosyslog" (don't write any messages to syslog),

### "noemail" (don't send any email messages), or "ALL" (to generate both

### syslog and email messages). "ALL" is the default. Both "nosyslog"

### and "noemail" can be combined with a comma to disable all logging

### and alerting.

ALERTING\_METHODS ALL;

### By default, psad checks for journalctl on systems where systemd is

### installed. If journalctl is running, then psad will automatically acquire

### syslog data from journalctl instead of parsing a file in the /var/log/

### directory.

AUTO\_DETECT\_JOURNALCTL Y;

### The following vars have psad acquire iptables log data from the

### /var/log/messages file which the local syslog daemon (usually) writes

### iptables log messages to. If the ENABLE\_SYSLOG\_FILE variable below is set

### to "N", then psad reconfigures syslog to write iptables log data to the

### /var/lib/psad/psadfifo fifo file where the messages are picked up by kmsgsd

### written to the file /var/log/psad/fwdata for analysis by psad. On some

### systems, having syslog communicate log data to kmsgsd can be problematic

### (syslog configs and external factors such as Apparmor and SELinux can play

### a role here), so leaving the ENABLE\_SYSLOG\_FILE variable set to "Y" is

### usually recommended. However, if psad is running on a system where systemd

### is installed and syslog messages are accessed via journalctl instead of

### being written to the filesystem, then the ENABLE\_FW\_MSG\_READ\_CMD

### functionality should be used to take over (see below).

ENABLE\_SYSLOG\_FILE Y;

IPT\_WRITE\_FWDATA Y;

IPT\_SYSLOG\_FILE /var/log/messages;

### This is primarily used to acquire syslog messages from journalctl on

### systems where systemd is running.

ENABLE\_FW\_MSG\_READ\_CMD N;

FW\_MSG\_READ\_CMD /bin/journalctl;

FW\_MSG\_READ\_CMD\_ARGS -f -k;

USE\_FW\_MSG\_READ\_CMD\_ARGS Y;

FW\_MSG\_READ\_MIN\_PKTS 30;

### When enabled, this instructs psad to write the "msg" field

### associated with Snort rule matches to syslog.

ENABLE\_SIG\_MSG\_SYSLOG Y;

SIG\_MSG\_SYSLOG\_THRESHOLD 10;

SIG\_SID\_SYSLOG\_THRESHOLD 10;

### Expect that all logged TCP SYN packets include the options portion of the

### TCP header (requires the --log-tcp-options argument to the iptables LOG

### rule). If a SYN packet is received that does not include TCP options, then

### it may be created by a scanner such as Eratta Security's "masscan"). Note

### that psad still does a check to see if at least one log message is seen

### includes the OPT field before expecting the remaining messages to also

### include this field.

EXPECT\_TCP\_OPTIONS Y;

### TTL values are decremented depending on the number of hops

### the packet has taken before it hits the firewall. We will

### assume packets will not jump through more than 20 hops on

### average.

MAX\_HOPS 20;

### Do not include any timestamp included within kernel logging

### messages (Ubuntu systems commonly have this)

IGNORE\_KERNEL\_TIMESTAMP Y;

### FIXME: try to mitigate the affects of the iptables connection

### tracking bug by ignoring tcp packets that have the ack bit set.

### Read the "BUGS" section of the psad man page. Note that

### if a packet matches a snort SID generated by fwsnort (see

### http://www.cipherdyne.org/fwsnort/)

### then psad will see it even if the ack bit is set. See the

### SNORT\_SID\_STR variable.

IGNORE\_CONNTRACK\_BUG\_PKTS Y;

### define a set of ports to ignore (this is useful particularly

### for port knocking applications since the knock sequence will

### look to psad like a scan). This variable may be defined as

### a comma-separated list of port numbers or port ranges and

### corresponding protocol, For example, to have psad ignore all

### tcp in the range 61000-61356 and udp ports 53 and 5000, use:

### IGNORE\_PORTS tcp/61000-61356, udp/53, udp/5000;

IGNORE\_PORTS NONE;

### allow entire protocols to be ignored. This keyword can accept

### a comma separated list of protocols. Each protocol must match

### the protocol that is specified in an iptables log message (case

### insensitively, so both "TCP" or "tcp" is ok).

### IGNORE\_PROTOCOL tcp,udp;

IGNORE\_PROTOCOLS NONE;

### allow packets to be ignored based on interface (this is the

### "IN" interface in iptables logging messages).

IGNORE\_INTERFACES NONE;

### Ignore these specific logging prefixes

IGNORE\_LOG\_PREFIXES NONE;

### Minimum danger level a scan must reach before any logging or

### alerting is done. The EMAIL\_ALERT\_DANGER\_LEVEL variable below

### only refers to email alerts; the MIN\_DANGER\_LEVEL variable

### applies to everything from email alerts to whether or not the

### IP directory is created within /var/log/psad/. Hence

### MIN\_DANGER\_LEVEL should be set less than or equal to the value

### assigned to the EMAIL\_ALERT\_DANGER\_LEVEL variable.

MIN\_DANGER\_LEVEL 1;

### Only send email alert if danger level >= to this value.

EMAIL\_ALERT\_DANGER\_LEVEL 1;

### Enable detection of malicious activity that is delivered via IPv6. If

### ip6tables is not logging any traffic, then psad won't know anything

### about IPv6, or this variable can be set to "N" (this would be slightly

### faster if ip6tables isn't logging anything).

ENABLE\_IPV6\_DETECTION Y;

### Treat all subnets on local interfaces as part of HOME\_NET (this

### means that these networks do not have to be manually defined)

ENABLE\_INTF\_LOCAL\_NETS Y;

### Include MAC addresses in email alert

ENABLE\_MAC\_ADDR\_REPORTING N;

### Look for the iptables logging rule (fwcheck\_psad is executed)

ENABLE\_FW\_LOGGING\_CHECK Y;

### Send no more than this number of emails for a single

### scanning source IP. Note that enabling this feature may cause

### alerts for real attacks to not be generated if an attack is sent

### after the email threshold has been reached for an IP address.

### This is why the default is set to "0".

EMAIL\_LIMIT 0;

### By default, psad maintains a counter for each scanning source address,

### but by enabling this variable psad will maintain email counters for

### each victim address that is scanned as well.

ENABLE\_EMAIL\_LIMIT\_PER\_DST N;

### If "Y", send a status email message when an IP has reached the

### EMAIL\_LIMIT threshold.

EMAIL\_LIMIT\_STATUS\_MSG Y;

### This variable is used to have psad throttle the email alerts it sends,

### and implemented as a per-IP threshold. That is, if EMAIL\_THROTTLE

### is set to "10", then psad will only send 1/10th as many emails for each

### scanning IP as it would have normally. All other variables also apply,

### so this throttle value is taken into account after everything else. The

### default of zero means to not apply any throttling.

EMAIL\_THROTTLE 0;

### If "Y", send email for all newly logged packets from the same

### source ip instead of just when a danger level increases.

ALERT\_ALL Y;

### If "Y", then psad will import old scan source ip directories

### as current scans instead of moving the directories into the

### archive directory.

IMPORT\_OLD\_SCANS N;

### syslog facility and priority (the defaults are usually ok)

### The SYSLOG\_FACILITY variable can be set to one of LOG\_LOCAL{0-7}, and

### SYSLOG\_PRIORITY can be set to one of LOG\_INFO, LOG\_DEBUG, LOG\_NOTICE,

### LOG\_WARNING, LOG\_ERR, LOG\_CRIT, LOG\_ALERT, or LOG\_EMERG

SYSLOG\_IDENTITY psad;

SYSLOG\_FACILITY LOG\_LOCAL7;

SYSLOG\_PRIORITY LOG\_INFO;

### Port thresholds for logging and -S and -A output.

TOP\_PORTS\_LOG\_THRESHOLD 500;

STATUS\_PORTS\_THRESHOLD 20;

### Signature thresholds for logging and -S and -A output.

TOP\_SIGS\_LOG\_THRESHOLD 500;

STATUS\_SIGS\_THRESHOLD 50;

### Attackers thresholds for logging and -S and -A output.

TOP\_IP\_LOG\_THRESHOLD 500;

STATUS\_IP\_THRESHOLD 25;

### Specify how often to log the TOP\_\* information (i.e. how many

### CHECK\_INTERVAL iterations before the data is logged again).

TOP\_SCANS\_CTR\_THRESHOLD 1;

### Send scan logs to dshield.org. This is disabled by default,

### but is a good idea to enable it (subject to your site security

### policy) since the DShield service helps to track the bad guys.

### For more information visit http://www.dshield.org

ENABLE\_DSHIELD\_ALERTS N;

### dshield.org alert email address; this should not be changed

### unless the guys at DShield have changed it.

DSHIELD\_ALERT\_EMAIL reports@dshield.org;

### Time interval (hours) to send email alerts to dshield.org.

### The default is 6 hours, and cannot be less than 1 hour or

### more than 24 hours.

DSHIELD\_ALERT\_INTERVAL 6; ### hours

### If you have a DShield user id you can set it here. The

### default is "0".

DSHIELD\_USER\_ID 0;

### If you want the outbound DShield email to appear as though it

### is coming from a particular user address then set it here.

DSHIELD\_USER\_EMAIL NONE;

### Threshold danger level for DShield data; a scan must reach this

### danger level before associated packets will be included in an

### alert to DShield. Note that zero is the default since this

### will allow DShield to apply its own logic to determine what

### constitutes a scan (\_all\_ iptables log messages will be included

### in DShield email alerts).

DSHIELD\_DL\_THRESHOLD 0;

### List of servers. Fwsnort supports the same variable resolution as

#### Snort.

HTTP\_SERVERS $HOME\_NET;

SMTP\_SERVERS $HOME\_NET;

DNS\_SERVERS $HOME\_NET;

SQL\_SERVERS $HOME\_NET;

TELNET\_SERVERS $HOME\_NET;

#### AOL AIM server nets

AIM\_SERVERS [64.12.24.0/24, 64.12.25.0/24, 64.12.26.14/24, 64.12.28.0/24, 64.12.29.0/24, 64.12.161.0/24, 64.12.163.0/24, 205.188.5.0/24, 205.188.9.0/24];

### Configurable port numbers

HTTP\_PORTS 80;

SHELLCODE\_PORTS !80;

ORACLE\_PORTS 1521;

### If this is enabled, then psad will die if a rule in the

### /etc/psad/signatures file contains an unsupported option (otherwise

### a syslog warning will be generated).

ENABLE\_SNORT\_SIG\_STRICT Y;

### If "Y", enable automated IDS response (auto manages

### firewall rulesets).

ENABLE\_AUTO\_IDS N;

### Block all traffic from offending IP if danger

### level >= to this value

AUTO\_IDS\_DANGER\_LEVEL 5;

### Set the auto-blocked timeout in seconds (the default is one hour).

### A value of 0 means block forever.

AUTO\_BLOCK\_TIMEOUT 3600;

### Set the auto-blocked timeout in seconds for each danger

### level - zero means to block permanently. Each of these

### can be set independently

AUTO\_BLOCK\_DL1\_TIMEOUT $AUTO\_BLOCK\_TIMEOUT;

AUTO\_BLOCK\_DL2\_TIMEOUT $AUTO\_BLOCK\_TIMEOUT;

AUTO\_BLOCK\_DL3\_TIMEOUT $AUTO\_BLOCK\_TIMEOUT;

AUTO\_BLOCK\_DL4\_TIMEOUT $AUTO\_BLOCK\_TIMEOUT;

AUTO\_BLOCK\_DL5\_TIMEOUT 0; ### permanent

### Enable regex checking on log prefixes for active response

ENABLE\_AUTO\_IDS\_REGEX N;

### Only block if the iptables log message matches the following regex

AUTO\_BLOCK\_REGEX ESTAB; ### from fwsnort logging prefixes

### Control whether "renew" auto-block emails get sent. This is disabled

### by default because lots of IPs could have been blocked, and psad

### should not generate a renew email for each of them.

ENABLE\_RENEW\_BLOCK\_EMAILS N;

### By setting this variable to N, all auto-blocking emails can be

### suppressed.

ENABLE\_AUTO\_IDS\_EMAILS Y;

### Enable iptables blocking (only gets enabled if

### ENABLE\_AUTO\_IDS is also set)

IPTABLES\_BLOCK\_METHOD Y;

### Specify chain names to which iptables blocking rules will be

### added with the IPT\_AUTO\_CHAIN{n} keyword. There is no limit on the

### number of IPT\_AUTO\_CHAIN{n} keywords; just increment the {n} number

### to add an additional IPT\_AUTO\_CHAIN requirement. The format for this

### variable is: <Target>,<Direction>,<Table>,<From\_chain>,<Jump\_rule\_position>, \

### <To\_chain>,<Rule\_position>.

### "Target": Can be any legitimate iptables target, but should usually

### just be "DROP".

### "Direction": Can be "src", "dst", or "both", which correspond to the

### INPUT, OUTPUT, and FORWARD chains.

### "Table": Can be any iptables table, but the default is "filter".

### "From\_chain": Is the chain from which packets will be jumped.

### "Jump\_rule\_position": Defines the position within the From\_chain where

### the jump rule is added.

### "To\_chain": Is the chain to which packets will be jumped. This is the

### main chain where psad rules are added.

### "Rule\_position": Defines the position where rule are added within the

### To\_chain.

###

### The following defaults make sense for most installations, but note

### it is possible to include blocking rules in, say, the "nat" table

### using this functionality as well. The following three lines provide

### usage examples:

#IPT\_AUTO\_CHAIN1 DROP, src, filter, INPUT, 1, PSAD\_BLOCK\_INPUT, 1;

#IPT\_AUTO\_CHAIN2 DROP, dst, filter, OUTPUT, 1, PSAD\_BLOCK\_OUTPUT, 1;

#IPT\_AUTO\_CHAIN3 DROP, both, filter, FORWARD, 1, PSAD\_BLOCK\_FORWARD, 1;

IPT\_AUTO\_CHAIN1 DROP, src, filter, INPUT, 1, PSAD\_BLOCK\_INPUT, 1;

IPT\_AUTO\_CHAIN2 DROP, dst, filter, OUTPUT, 1, PSAD\_BLOCK\_OUTPUT, 1;

IPT\_AUTO\_CHAIN3 DROP, both, filter, FORWARD, 1, PSAD\_BLOCK\_FORWARD, 1;

### Flush all existing rules in the psad chains at psad start time.

FLUSH\_IPT\_AT\_INIT Y;

### Prerequisite check for existence of psad chains and jump rules

IPTABLES\_PREREQ\_CHECK 1;

### Enable tcp wrappers blocking (only gets enabled if

### ENABLE\_AUTO\_IDS is also set)

TCPWRAPPERS\_BLOCK\_METHOD N;

### By default, enable whois lookups against scanning IP addresses.

ENABLE\_WHOIS\_LOOKUPS Y;

### Set the whois timeout

WHOIS\_TIMEOUT 60; ### seconds

### Set the number of times an ip can be seen before another whois

### lookup is issued.

WHOIS\_LOOKUP\_THRESHOLD 20;

### Use this option to force all whois information to contain ascii-only data.

### Sometime whois information for IP addresses in China and other countries

### can contain non-ascii data. If this option is enabled, then any non-

### ascii characters will be replaced with "NA".

ENABLE\_WHOIS\_FORCE\_ASCII N;

### This variable forces all whois lookups to be done against the source IP

### even when they are associated with a directly connected local network. IT

### is usually a good idea to leave this setting as the default of 'N'.

ENABLE\_WHOIS\_FORCE\_SRC\_IP N;

### By default, enable reverse DNS lookups against scanning IP addresses.

ENABLE\_DNS\_LOOKUPS Y;

### Set the number of times an ip can be seen before another dns

### lookup is issued.

DNS\_LOOKUP\_THRESHOLD 20;

### Enable psad to run an external script or program (use at your

### own risk!)

ENABLE\_EXT\_SCRIPT\_EXEC N;

### Define an external program to run after a scan is caught.

### Note that the scan source ip can be specified on the command

### line to the external program through the use of the "SRCIP"

### string (along with some appropriate switch for the program).

### Of course this is only useful if the external program knows

### what to do with this information.

### Example: EXTERNAL\_SCRIPT /path/to/script --ip SRCIP -v;

EXTERNAL\_SCRIPT /bin/true;

### Control execution of EXTERNAL\_SCRIPT (only once per IP, or

### every time a scan is detected for an ip).

EXEC\_EXT\_SCRIPT\_PER\_ALERT N;

### Enable psad to run an external script or program upon setting

### iptables block (use at your own risk!)

ENABLE\_EXT\_BLOCK\_SCRIPT\_EXEC N;

### Define an external program to run after a scan is blocked.

### Note that the scan source ip can be specified on the command

### line to the external program through the use of the "SRCIP"

### string (along with some appropriate switch for the program).

### Of course this is only useful if the external program knows

### what to do with this information.

### Example: EXTERNAL\_BLOCK\_SCRIPT /path/to/script --ip SRCIP -v;

EXTERNAL\_BLOCK\_SCRIPT /bin/true;

### Some syslog daemons support a customized time stamp, so allow

### a user-specified regex to account for this when necessary (disabled

### by default). This regex should also extract the hostname from the

### syslog messages as well, so the timestamp and the hostname should be

### stored in $1 and $2 respectively. Here is an example syslog message

### and how to extract the timestamp and hostname:

### 2015-03-08T02:25:11.444012+02:00 servername kernel: ...

### ...would be handled properly with:

### ^\s\*([\d\-]+T(?:\d{2}\:){2}\d{2}\S+)\s+(\S+)\s+kernel:

ENABLE\_CUSTOM\_SYSLOG\_TS\_RE N;

CUSTOM\_SYSLOG\_TS\_RE ^\s\*((?:\S+\s+){2}\S+)\s+(\S+)\s+kernel\:;

### Disk usage variables

DISK\_CHECK\_INTERVAL 300; ### seconds

### This can be set to 0 to disable disk checking altogether

DISK\_MAX\_PERCENTAGE 95;

### This can be set to 0 to have psad not place any limit on the

### number of times it will attempt to remove data from

### /var/log/psad/.

DISK\_MAX\_RM\_RETRIES 10;

### Enable archiving of old scan directories at psad startup.

ENABLE\_SCAN\_ARCHIVE N;

### Truncate fwdata file at startup

TRUNCATE\_FWDATA Y;

### Only archive scanning IP directories that have reached a danger

### level greater than or equal to this value. Archiving old

### scanning ip directories only takes place at psad startup.

MIN\_ARCHIVE\_DANGER\_LEVEL 1;

### Email subject line config. Change these prefixes if you want

### psad to generate email alerts that say something other than

### the following.

MAIL\_ALERT\_PREFIX [psad-alert];

MAIL\_STATUS\_PREFIX [psad-status];

MAIL\_ERROR\_PREFIX [psad-error];

MAIL\_FATAL\_PREFIX [psad-fatal];

### URL for getting the latest psad signatures

SIG\_UPDATE\_URL http://www.cipherdyne.org/psad/signatures;

### These next two are psadwatchd vars

PSADWATCHD\_CHECK\_INTERVAL 5; ### seconds

PSADWATCHD\_MAX\_RETRIES 10;

### Directories

INSTALL\_ROOT /;

PSAD\_DIR $INSTALL\_ROOT/var/log/psad;

PSAD\_RUN\_DIR $INSTALL\_ROOT/var/run/psad;

PSAD\_FIFO\_DIR $INSTALL\_ROOT/var/lib/psad;

PSAD\_LIBS\_DIR $INSTALL\_ROOT/usr/lib/psad;

PSAD\_CONF\_DIR $INSTALL\_ROOT/etc/psad;

PSAD\_ERR\_DIR $PSAD\_DIR/errs;

CONF\_ARCHIVE\_DIR $PSAD\_CONF\_DIR/archive;

SCAN\_DATA\_ARCHIVE\_DIR $PSAD\_DIR/scan\_archive;

ANALYSIS\_MODE\_DIR $PSAD\_DIR/ipt\_analysis;

SNORT\_RULES\_DIR $PSAD\_CONF\_DIR/snort\_rules;

FWSNORT\_RULES\_DIR /etc/fwsnort/snort\_rules; ### may not exist

### Files

FW\_DATA\_FILE $PSAD\_DIR/fwdata;

ULOG\_DATA\_FILE $PSAD\_DIR/ulogd.log;

FW\_CHECK\_FILE $PSAD\_DIR/fw\_check;

DSHIELD\_EMAIL\_FILE $PSAD\_DIR/dshield.email;

SIGS\_FILE $PSAD\_CONF\_DIR/signatures;

PROTOCOLS\_FILE $PSAD\_CONF\_DIR/protocols;

ICMP\_TYPES\_FILE $PSAD\_CONF\_DIR/icmp\_types;

ICMP6\_TYPES\_FILE $PSAD\_CONF\_DIR/icmp6\_types;

AUTO\_DL\_FILE $PSAD\_CONF\_DIR/auto\_dl;

SNORT\_RULE\_DL\_FILE $PSAD\_CONF\_DIR/snort\_rule\_dl;

POSF\_FILE $PSAD\_CONF\_DIR/posf;

P0F\_FILE $PSAD\_CONF\_DIR/pf.os;

IP\_OPTS\_FILE $PSAD\_CONF\_DIR/ip\_options;

PSAD\_FIFO\_FILE $PSAD\_FIFO\_DIR/psadfifo;

ETC\_HOSTS\_DENY\_FILE /etc/hosts.deny;

ETC\_SYSLOG\_CONF /etc/syslog.conf;

ETC\_RSYSLOG\_CONF /etc/rsyslog.conf;

ETC\_SYSLOGNG\_CONF /etc/syslog-ng/syslog-ng.conf;

ETC\_METALOG\_CONF /etc/metalog/metalog.conf;

STATUS\_OUTPUT\_FILE $PSAD\_DIR/status.out;

ANALYSIS\_OUTPUT\_FILE $PSAD\_DIR/analysis.out;

INSTALL\_LOG\_FILE $PSAD\_DIR/install.log;

### PID files

PSAD\_PID\_FILE $PSAD\_RUN\_DIR/psad.pid;

PSAD\_FW\_READ\_PID\_FILE $PSAD\_RUN\_DIR/psad\_fw\_read.pid;

PSAD\_CMDLINE\_FILE $PSAD\_RUN\_DIR/psad.cmd;

KMSGSD\_PID\_FILE $PSAD\_RUN\_DIR/kmsgsd.pid;

PSADWATCHD\_PID\_FILE $PSAD\_RUN\_DIR/psadwatchd.pid;

### List of ips that have been auto blocked by iptables

### or tcpwrappers (the auto blocking feature is disabled by

### default, see the psad man page and the ENABLE\_AUTO\_IDS

### variable).

AUTO\_BLOCK\_IPT\_FILE $PSAD\_DIR/auto\_blocked\_iptables;

AUTO\_BLOCK\_TCPWR\_FILE $PSAD\_DIR/auto\_blocked\_tcpwr;

### File used internally by psad to add iptables blocking

### rules to a running psad process

AUTO\_IPT\_SOCK $PSAD\_RUN\_DIR/auto\_ipt.sock;

FW\_ERROR\_LOG $PSAD\_ERR\_DIR/fwerrorlog;

PRINT\_SCAN\_HASH $PSAD\_DIR/scan\_hash;

### /proc interface for controlling ip forwarding

PROC\_FORWARD\_FILE /proc/sys/net/ipv4/ip\_forward;

### Packet counters for tcp, udp, and icmp protocols

PACKET\_COUNTER\_FILE $PSAD\_DIR/packet\_ctr;

### Top scanned ports

TOP\_SCANNED\_PORTS\_FILE $PSAD\_DIR/top\_ports;

### Top signature matches

TOP\_SIGS\_FILE $PSAD\_DIR/top\_sigs;

### Top attackers

TOP\_ATTACKERS\_FILE $PSAD\_DIR/top\_attackers;

### Counter file for Dshield alerts

DSHIELD\_COUNTER\_FILE $PSAD\_DIR/dshield\_ctr;

### Counter file for iptables prefixes

IPT\_PREFIX\_COUNTER\_FILE $PSAD\_DIR/ipt\_prefix\_ctr;

### iptables command output and error collection files; these are

### used by IPTables::ChainMgr

IPT\_OUTPUT\_PATTERN psad\_iptout.XXXXXX;

IPT\_ERROR\_PATTERN psad\_ipterr.XXXXXX;

### system binaries

iptablesCmd /sbin/iptables;

ip6tablesCmd /sbin/ip6tables;

shCmd /bin/sh;

wgetCmd /usr/bin/wget;

gzipCmd /bin/gzip;

mknodCmd /bin/mknod;

psCmd /bin/ps;

mailCmd /bin/mail;

sendmailCmd /usr/sbin/sendmail;

ifconfigCmd /sbin/ifconfig;

ipCmd /sbin/ip;

killallCmd /usr/bin/killall;

netstatCmd /bin/netstat;

unameCmd /bin/uname;

whoisCmd $INSTALL\_ROOT/usr/bin/whois\_psad;

dfCmd /bin/df;

fwcheck\_psadCmd $INSTALL\_ROOT/usr/sbin/fwcheck\_psad;

psadwatchdCmd $INSTALL\_ROOT/usr/sbin/psadwatchd;

kmsgsdCmd $INSTALL\_ROOT/usr/sbin/kmsgsd;

psadCmd $INSTALL\_ROOT/usr/sbin/psad;