This is all you should need to know. If you are having problems, you are a dummy and need to read this doc.

Basically if you have 3 employees, 1 site and an AD server and the domain admin password, its brainless. Any more than 3 employees and 1 site then you will need therapy. The deal is unless you are an AD/LDAP brain machine, you will be clueless debugging a multi-site environment. There are so many components that you will never know where the real bug is. Not totally a CP problem, but just life in general. One downside is that CP’s GUI wizards, GUI and debugging tools are just OK, not great. You can tell they have been written by 100 developers/20 documenters over 10 years, so the flow is not the best and the documentation is OK. Getting a lot better, but not close to an iPhone. Perhaps a BlackBerry.

If you read this SK

The intent of this document is to tie some loose ends together for dummies like me, especially for enterprise sites.

Identify Awareness Overview
Why do I need Identity Awareness (IA)??

The world is going mobile and so DHCP is going to be the rule. How do you build a rulebase with dynamic IP addresses?

Protocols are being developed daily and DCOM/RPC and port 80 are the ships sailing the digital seas. How do you filter a protocol when its jammed into port 80 with 100 other protocols?

Welcome to the world of Nex Gen firewalls, Identity Awareness is the glue.

a. Used for tracking who is doing what in the logs:
b. Or to regulate what users can access what resources:

And even go so far as to force users to authenticate at the firewall in order to access their applications.
In this document we will show one setup where a firewall is protecting a site and requires users to authenticate in order to access any server in the site.

Assume an Active Directory environment with built in LDAP services is the environment that stores user databases and tracks user events.

**AD and LDAP Tree**
Active directory has a forest of domains. A forest is an administrative boundary, a trusted group of admins for the same org administer the whole environment. For example, a company called abc.com. But abc.com is so huge that it's broken into sub-domains with sub-admins. Each domain contains a list of users local to that domain. A forest allows domains to build trust amongst them in order to allow searching, authentication, authorization, groups, etc between all the domains.

Internal to a domain are folders (just like your file system) that LDAP usually calls Organizational Units OU. So the DC's are physical machines and the OU's are folders in the DC's and the domain.

NOTE: The LDAP labels DC, OU are not MS or LDAP standards just guidelines. We could call them anything like XX or YY. But MS uses DC for domain controller and OU for folder.
Each domain has one or more domain controllers (DC). These are physical machines that share user data for that domain and track user events. They are usually located in the local offices so that AD traffic and replication data does not clog small WAN pipes. So in this case hq.abc.com has 2 domain controllers at the Boston and SF offices. They replicate between them the users for hq and no other domain.
AD is huge. If all the DC’s in abc.com replicated to each other the whole thing would crash. BUT, hq.abc.com still has to send email to sales.abc.com. So there is about 10% of the global environment that is replicated to all abc.com DC’s. This is called Global Catalog (GC). There is just enough information in the GC to send email, login, search for user info, start video chats, etc.

This is where AD will bite you: Groups.
Groups are important for scalable solutions. In a rulebase you don't want to enter 1000 people into the rule, you want to always use groups. So in scalable solutions, groups so just magically work or the software should prevent you from wandering into the danger zone.

AD has several different types of groups.

**Group Scope**

If you have a user JoeBlow that is in a hierarchy of groups across domains or in Universal group in another domain... good luck. CP (and maybe all 3rd party products) will have problems with these depending on your configuration. We'll talk more about this later.

UserJoeBlow->Group1Mgrs_HQ.ABC.COM->Group2SalesNotifications_SALES.ABC.COM
UserJoeBlow->GlobalMgrs_HQ.ABC.COM->UniversalMgrsGroup_ ABC.COM

<UniversalGroup>

**LDAP**

External to AD (non-Microsoft apps), LDAP is used to gather information about the forest. Examples:

1. Identify person/machine
2. Authenticate person/machine
3. What groups does person belong to

AD has an LDAP server on it that external applications can use to parse the AD tree. There are tools that will do this for you:

1. Active Directory Administrative Center (our choice. MS default)
Here are some tips to install it.

Remove Server Administrative Tools

Once installed you can start doing searches and look for people and see all their LDAP links. Make sure you do Global Searches and then work into the domains from there.
In order for CP IA to interact with AD, you need an AD service account with pseudo-admin privs. Read this: [AD User with WMI Rights](#): SK93938. The account needs these privs:

1. Search AD
2. Get AD group info
3. Receive Windows Management Interface (WMI) events like logins, logoffs

Let’s go looking for this account.
Global search for pseudo admin user

Here it is!!! Look at the LDAP links
You might want to save this info, you will definitely need it to hook CP into AD. For fun start use ADAC to browse through the DCs and OUs to find your user info. This is left as Exercise for the student. Good Luck!
CheckPoint Identity Awareness Components

These are all the CP components that use Identity Awareness (IA).

Smart Dashboard IA Components

The link between CP and IA is done with **LDAP Account Units or AUs**. Think of the AU as a proxy or protocol converter between CP and LDAP/AD. Doesn’t do any filtering, just grabs data from
LDAP and presents to CP.

Here is where the wheels fall off the wagon at enterprise shops. It will be hard for CP admins to hook up with AD admins to make this work. Unless these people are drinking buddies or their bosses are husband/wife, husband/husband, wife/wife, etc trying to get the right admin account, figure out all the LDAP magic, try to get IA to work correctly with groups/UniversalGroups, slow WAN links, slow GUI response, navigate the right domain trees, and then trying to debug both AD and CP at the same time. You will need lots of therapy and blue pills before it all works. The bigger the AD environment, the more blue pills.

If by some luck you get a hold of the AD super domain account that never expires, then you are golden no/less worries.

So first thing is HOW MANY AUs do you need? First pop a couple blue pills and hum “Namaste” a couple times.

CP claims you can only have 1 AU per AD domain. Of course there might be 1000 DCs in the one AD domain no problem, but only 1 AU. Read this:
SK92782: AD Forest Support – Only 1 AU per domain or group searches will fail
sk60301: ADQuery Details – Performance info
sk88520: PDP Large Setups – Performance tips

FYI: we have setup about 40 AUs going to 500’ish DCs in 1 domain with 1 AU per site and no issues so far. So not sure what CP says is true. I have several tips on scaling in the following sections. One thing is to make sure you use that same color and object name prefix for all IA/AD network objects or else you will never be able to search and find what you need. See pic below.

FYI: our main DCs have tens of thousands of employees on them. IA WMI traffic to a subscribed firewall is in the gigs per minute, so you might want to chain firewall PDP/PEP together.

So the answer is.....I’m not sure what to tell you. I like to have 1:1 for a reason you will find out at the end...The magic is hidden #1001 DC priority and having to reset all the DC priorities as you add in new DCs. Doesn’t make sense right now but it will.
NOTE: dynamic group membership not working right

LDAP Account Unit

OnSite AD/LDAP Servers

etc

etc

Note that all the AUs (also AD objects) are same color and same prefix. VERY IMPORTANT!!! For large sites. More later

Use LDAP AU to setup connection to AD servers for LDAP data or AD WMI events. CP SK's say to use only 1 LDAP AU for all your AD servers, but that is BS ignore it. Easier to do this...We'll show you why later.

NOTE: dynamic group membership not working right

LDAP setup tips

Very important info:

a. Make sure the AU prefix name and color are all the same. Same with AD network object. Save you tons of search time.

b. AD Domain you want to search through. Make sure you don't search through the whole forest just the domain relevant to the site. So get the right domain name. i.e. hq.abc.com vs abc.com

c. User name with WMI privs that can at least receive WMI events

d. LONG LDAP user name of above...you can get the long name using the LDAP Active Directory Command Center
Next we have to point to the DC or DC's of the AD domain. This will allow CP's DMS, Gateway, DLS to receive WMI events and perform LDAP queries to a specific DC. This is kinda a crux piece of the design. Here you can specify 1 or 1000000 DCs, and it may come back to haunt
you either way. Too few and you miss tagging log events, cross domain overlap on LDAP queries which screw up group memberships. Too many and you will flood WAN links, overload DLS servers, will be really really hard to add in additional DCs with the right priorities. This screen is the art that large enterprises will have to figure out.

We have been using 1 AU per DC because it gives us finer granularity over which FW goes to which DC and adding DCs is much much easier. It has been working even though CP says it won’t because group searches should fail. Lately I have been thinking about using 1 UA for an AD domain (as recommened), but I’ll have to dbedit some priority fields to make it work.

Read on.

I am using these creds as the AD admin user with WMI rights.

CN=AD_admin,OU=UTM_Identity,OU=Service Accounts,DC=hq,DC=abc,DC=com

Of course none of this will work the first-10th time, so try using this tool to make sure your credentials are solid:
I once typed in the wrong password. Took me 4 days to figure it out, kept locking out the account and had this ripple effect through the AUs. Unfortunately a 10 minute AD lockout timer would expire so it was then hard to tell debug something that seemingly randomly failed but now suddenly worked…until another AU locked up….repeat.

SOOOO make sure the pwd is correct!!!

BACK to LDAP AU config
If you click this make sure it does NOT change the tree name or you will slow down searches searching under some supertree.

Make sure this is the tree you are searching under.

Might want to up this to 10,000 or so but will slow down retrievals a bit in the GUI.
SmartDashboard

So let’s see how SmartDashboard works with IA.

**Smart Dashboard IA Components**

Obviously SmartDashboard configures the whole IA business.

The other IA component is

**The Picker**

The Picker is the software that allows you to pick users/machines to put into the rulebase.
It gathers information during an interaction between your desktop and the LDAP server. Which LDAP server???
Well, wish I could tell you but no clue at this point. I think?? It will go to the LAST DC you configured for the DMS domain, but I can’t confirm. Not sure what the “assign entity to” is for either, just showed up in 77.10 and no documentation on it. Pretty cool huh? (blue pill).

(UPDATE: 2/26/15: There might be a GUI fix for this check into it. Gui 4702)
If you have a single AU in a single AD domain, ignore this part

When your picker chooses an access role, the picker will label which AU is associated with that role and store it on the firewall.

So here we are creating a group US-HQ-Admins:

And it will eventually be stored in the firewall tagged with the AU it was found in: Siberia:
Why do you care???

Because when the firewall is trying to tie the USER to this group US_HQ_Admins, the firewall will make sure the LDAP query it makes to an AU matches the AU in the group US_HQ_Admins. In this case they both have to be to SIBERIA. If for some reason they are different (multiple AUs in a domain)…..SOL pal.

How to fix this??

Welllllllllll. Get ready.

Add this line into CPprofile.sh and reboot/cpstop/start.
And the firewall will ignore the AU match and just match up the access role to the LDAP group regardless of what AU it is from.

Pretty cool HUH?

Slow Picker

So if your picker is slow or doesn’t work, then you need to go back to Chapter 1 and read: IA is easy to deploy and scalable.

Here are some debug tips

1. New hotfix came out for Dashboard Oct 2014. They were trying to sort 1 billions of entries and it took forever

2. You might be going to Siberia DC, check which DC it is going to in the GUI or you might have to sniff port 389 from your PC
3. You could reduce # of entries retrieved

4. Make sure it works with wbemtest.exe, the tool I showed above

Otherwise I’m not really sure but thank god its easy to deploy and scalable.

Now I’m not going to talk about access roles. You can read about them in the admin guide. Just remember they are once again proxy’s between LDAP and the rule base. You can’t specify a person directly, you have to reference an Access Role with a person embedded within it. Tomato-Tomaaaaato. You can also group people into a single access role so it kinda works like groups too.
Let’s assume the DMS is NOT setup for IA. Hey guess what it still does LDAP queries.
So to be honest not sure why this is..but it is. So just FYI in case you are debugging you know what is making LDAP queries from where.

On the other hand, lets turn ON IA at the DMS/DLS.
Oh oh, what happened???

Well, beware that now ALL!!! WMI events from ALL DCs in the world are heading your way. Well I exaggerate a bit. To be more exact, for THIS domain IF you setup an AU(s) with all the DCs in your org, then yes it will soon be WMI-fest in DMS-ville. Shields up. Once again, no problem in 3 person 1 site shop, big problem in enterprise shop.
DLS – Don’t flood DLS server with WMI

Why would you do this to yourself? Well, for logging you want to track users/log event and WMI will fill in the blanks.

1. If you are doing IA at the firewall, then you probably don’t have to turn IA on at the DLS/DMS. The firewalls insert user/group into log entries for you with their IA.

2. Because firewalls may only point to 1 local DC of 10000 enterprise DCs, log entries will only have identities for that 1 local DC. So log entries will be incomplete because the firewall can only fill in user names of the local users and not enterprise users.
You have to enable IA at the DMS/DLS in order for the log servers to get WMI events from the whole enterprise to fill in usernames.
As stated at the very start (reread), Firewalls are configured with Identity Awareness for several purposes.

1. Making access decisions with USERS instead of IPs in the rulebase
2. Tagging log entries with user names

And IA is the tool to do that. When you enable IA, you now have the capability to perform the above two functions:
When you enable IA, Firewalls register with AD to receive Windows Management Instrumentation (WMI) events and perform LDAP queries.

WMI events are used to:

1. Correlate IP addresses with user names
2. Provide a relative escalation of trust that the user/machine identified in the WMI event has been authenticated by AD and is local to your AD environment.
3. Tag log entries with user names
4. A keepalive system to understand if the user is still logged in to their desktop

This is how you register a firewall to receive WMI events:
WARNING: Remember this question, how many AUs should you have?? The above will have all DCs in the hq_abc_AU sending WMI events to the firewall. Now beware that if there are multiple/100 DCs, then you might be bogging down the WAN line with WMI from 100 DCs.

However!

If you had 1 DC per AU, then only 1 DC would be sending WMI to the firewall. (BUT, CP claims this doesn’t work)

So pick your poison.

NOTE: You can limit this somewhat in the ‘Advanced’ tab and by ignoring WMI for machine identities.
Firewalls can also perform LDAP queries on the specified LDAP server.

1. LDAP search: Gather AD group information about the user so the firewall can make Access Decisions. “Is user X in group Finance_Users_Role access group (see below)”
2. Tag log entries with AD user/machine names

3. LDAP bind. Definitively authenticate the user with AD to absolutely prove the user X on machine 1.1.1.1 is known to AD because user X has the correct credentials.
Firewall to LDAP binding is done here:

By doing this you are linking the firewall to LDAP. Once again, look out. If the AU has 1000 DCs in it….not sure which DC it will use.

NOTE: This AU configuration menu is for Captive Portal LDAP only. Not ADQuery/WMI

There is a little magic you can try in order to limit DC scope creep. I’ve only use this a bit, so not sure about all the corner cases. I’ve been told this is the best way to do it and it supercedes what is in the IA tab. A developer said that this is the default in case nothing else is set and ALL the blades on the gateway (AB, DLP, App control) use this to perform their IA lookups.
This will link the firewall with LDAP, but allow you to specify what DC it can reach.

Now let's talk priority field (get ready)
DANGER WILL ROBINSON

If an AD domain has 1000000000 DCs and you decide to create 1 AU with 10000000000 DCs inside of it (above picture) great for you (and you followed CP’s mantra, Medallion of Honor to you). (blue pill). When you push policy to your first firewall, prepare yourself to receive WMI events from 10000000000 DCs no matter WHAT priority you set here. (blue pill).

Know what’s even funnier? Let’s say you have 1000 firewalls with IA deployed. One of your local Microsoft yahoos decides to add another DC at another new site in this same AD domain. Well there, get ready to go through all 1000 firewalls and update the priority fields one-by-one or
else the new DC will send WMI events to all 1000 firewalls. Cute huh? Easy to deploy and scalable. (blue pill).

Now the LDAP queries will go to first 1 priority than 2 priority, etc until it gets a response. But for WMI, you are at the end of a firehose.

SOOOOOO perhaps the priority should be labelled “LDAP priority” instead of “Default priority”. Or maybe 2 priorities for LDAP and WMI? Yeah I know, a foolish thought. Too obvious for easy to deploy and scalable solutions.

Good news is there is a super secret 1001 priority that enables the firewall to ignore DCs completely. BUT, in this screen you CANNOT enter 1001. Cute huh? You have to go to here to enter it.
NOTE: This is for WMI and LDAP both. You effectively kill IA on the firewalls by doing this. SmartDashboard STILL makes LDAP queries from you workstation, but the firewalls will not.

There is some dbedit magic to set the NEW DCs to 1001 so that ALL 1000 firewalls will ignore the new DC. TBD.

[dbedit TBD]

SUMMARY: In theory (and CP preference) is you have 1 AU for 1 MS AD domain and put all your 10000000 DCs into this AU, like below. The ‘other’ tab in theory will be what the firewall uses to probe DCs. In theory this ‘other’ setup will override all your other settings. I said in theory because I am able to break this theory but you’ve taken so many blue pills by now I don’t want to spoil your high with more complexity.

PROBLEM: I suggest you disable DCs you don’t want to go to, but you are going to have a lot of work on your hands so not sure what to tell you Vern. Working on dbedit script, please hold while I service your request.
CheckPoint says never have multiple AUs for the same AD domain in the same DMS. I’m saying….. If you have multiple AUs for a single AD domain. NEVER put multiple AUs from the same AD domain into the same gateway configuration as below:
My testing has shown that the gateway will get confused and go to the wrong DCs.

If you have multiple AD domains in your DMS domain, and a user belongs to multiple groups in multiple cross domains, then you are on your own sorry. Not sure what will happen.

**Getting Dicey – Cross AD domain groups**

SK92782: [AD Forest Support](#) – Only 1 AU per domain or group searches will fail

Scalable solutions usually mean that as you add more entities into the mix, the model retains its integrity. Unfortunately AD is not that flexible when working with IA. Cross-AD-domain groups don’t always work right. Theoretically if CP allows you to have multiple AUs, and there is no restriction in the software on how they are used then CP should support cross-domain group support. Well we are not at that point in life so suck it up people.

UserJoeBlow->Group1Mgs_HQ.ABC.COM->Group2SalesNotifications_SALES.ABC.COM

Is when problems start appearing.
If IA/AppControl/Captive Portal are heavily used in your org, you may want to consider aligning your DMS domains with your AD domains. Not perfect but may work for your org.

Another idea that I have seen used in multiple orgs is no cross domain usage by users. If UserX.flowers.com wants to do something in the Wineries domain, then they are given an additional account UserX.Wineries.com. Problem solved. These are mostly support people.

I’m not sure this will scale, but hopefully the IA<>CP<>AD integration will become much tighter over time.

Work In Progress......

Read on.
CheckPoint Processes

CP has 3 processes that are responsible for monitoring IA:

Adlog – Keeps tracks of WMI events from the LOCAL AD Domain controller. Builds a database of User to machine IP mappings based on WMI events from AD (login to you laptop, use MS services like mount a file share, RDP to terminal server, use IE to log into IIS web server).

ADLOG is like an external cache of WMI events for AD. Because an IP is in this cache doesn’t mean the firewall cares or uses the IP addresses. The cache is there so that PDP and PEP can refer to the IP list in the future when needed.

This cache is used for a keepalive monitoring to see if users are still logged in. The hassle with AD is that there are NO LOGOUT events, so all 3rd party products must guess if the user is still logged in. HOW??? By seeing if the user is logging in, sharing files, browsing MS web sites, locking their screens, etc. Then timers are set and the software does a “Geez, I have not seen any events from JOE’s IP for 12 hours, perhaps he logged out”.

This obviously has huge problems. KIOSKS: If Joe logs in for 5 minutes he is then trusted for 12 hours. If Joe logs out in 5 minutes and FRAN logs in, now Joe and FRAN
will be associated with the same IP for 12 hours or so. So the access rules work the same for FRAN as they do Joe, even if JOE is the only one in the access rule.

Low lease time DHCP and roaming Mobile environments are especially fun to deal with. There are ways to mitigate this…maybe in my next long dreary blog.

Anyways…………..

NOTE: Adlog keeps track of AD registered MACHINES and AD registered USERS. Note that MACHINES can be identified by IP addresses and/or by their existence in AD. An IP address in North Korea probably has less integrity then an IP address registered in your AD. So when Picker lists MACHINES, it is listing machines with some level of trust because they are joined into your domain and NOT just random IP addresses.
What do these tables look like?

This command `adlog a dc` summarizes the table:

```
[Expert8] # if adlog a dc
Domain controllers: 
Domain Name       IP Address                      Connection state  Events in the last hour
hq.abc.com       123.123.123.123                      has connection:  $3950
Ignored domain controllers on this gateway:
No ignored domain controllers found.
```

Note there may be more than one user on an IP. Could be a terminal server, could be a Kiosk and UserA has not timed out.
<table>
<thead>
<tr>
<th>IP</th>
<th>Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1.1.1</td>
<td><a href="mailto:user1@abc.com">user1@abc.com</a>; <a href="mailto:user2@abc.com">user2@abc.com</a>; <a href="mailto:user3@abc.com">user3@abc.com</a></td>
</tr>
<tr>
<td>2.2.2.2</td>
<td><a href="mailto:user4@abc.com">user4@abc.com</a>; <a href="mailto:user5@abc.com">user5@abc.com</a>;</td>
</tr>
<tr>
<td>3.3.3.3</td>
<td>etc</td>
</tr>
<tr>
<td>4.4.4.4</td>
<td>etc</td>
</tr>
<tr>
<td>etc</td>
<td>etc</td>
</tr>
</tbody>
</table>
1. Note these associations are kept for a minimum of 12 hours.
Here are 2 users logged into same system so same IP address:

Session: bc384c9c
Session UUID: {754043E5-80C4-F72D-68A2-E3B634BEC2B}
Ip: 123.123.123.123
Users:

user1@hq.abc.com  (b25c81e9)
  Groups: All users
  Roles: Any DROP
  Client Type: AD Query
  Authentication Method: Trust
  Connect Time: Tue Dec 23 17:52:27 2014
  Next Reauthentication: Wed Dec 24 05:52:57 2014
  Next Connectivity Check: Wed Dec 24 05:52:57 2014

user2@hq.abc.com  (e9111918)
  Groups: All Users
  Roles: Any DROP, Group_Auth_Group
  Client Type: AD Query
  Authentication Method: Trust
  Connect Time: Tue Dec 23 17:48:42 2014
  Next Reauthentication: Wed Dec 24 05:49:12 2014
  Next Connectivity Check: Wed Dec 24 05:52:57 2014

Packet Tagging Status: Not Active
Published Gateways: Local

---

Various applicable scenarios:

- **New Association**: In case this IP address was unknown before this association, the association is added to the database with an expiry time of \((\text{now} + \text{configured timeout})\) (by default, the configured timeout is 12 hours).

- **Existing Association**: If this Association for this IP address was known before, the expiry time is updated to \((\text{now} + \text{configured timeout})\). (The configured timeout is 12 hours)

- **Machine change**: If the machine has changed, all of the currently known user associations are revoked, as well as the current associated machine, and the new machine is added to the database with an expiry time of \((\text{now} + \text{configured timeout})\).

- **New User**: If an unknown user association is encountered, and "assume one user per IP" is "off" (default), the new association is added with an expiry time of \((\text{now} + \text{configured timeout})\). (The configured timeout is 12 hours).

- **User Change**: If an unknown user association is encountered, and "assume one user per IP" is "on", all of the currently associated users are revoked, and the new association is added as the only user for this IP address. If there were any machine associations for this IP address, they are left intact. See "Single User Assumption" in the Identity Awareness Administration Guide for more information.

- **Multi user host detected**: If 7 (by default) users are currently associated for the same IP address, the IP address is automatically considered a "multi user host". A log about it is issued, all of the currently associated users are revoked and all new user associations for this IP address are ignored.
Here are some WMI events the firewall follows.

a. Receive Windows Windows Instrumentation (WMI) Events (like login, logout, etc)
   On 2003 Domain controllers the events are 672, 673 and 674
   On 2008 Domain controllers the events are 4624, 4768, 4769 and 4770

b. 4624: Local login

c. 4768: Kerberos Ticket was requested from DC. This identifies the user/machine in Kerberos

d. 4769: Kerberos Service Ticket was requested: User/machine requires access to resource like file system mapping or web server

e. 4770: Kerberos Service was renewed: Kerberos will auto renew service ticket if user still logged in

Notice no LOGOUT events. This is why ADLOG is a keepalive database.

You can see WMI events in the logs with these events. NOTE: Look for entries with “Identity Source: AD Query”. These are WMI events. Don’t confuse them with “Captive Portal” Login events, those are kept in the PDP database.
Captive Portal events not the same as AD Query events. They are generated by the firewall captive portal and NOT WMI
Debug ADLOG

Here are the basic debug commands you cannot forget:

1. `adlog a dc` – find out if connected to dc and getting events
2. `adlog a query all` – dump IP/user table
3. Here is another debug tool that I have not played with but shows much promise:
   
   `adlogconfig`
NOTE: FYI: adlog is the 'wmic' service that only runs on the active member of the cluster.

Standby cluster member does not run adlog. Adlog tables are not replicated on standby member, but pdp/pep tables are
sk102986: IA Debug Tips
sk102027: Clear/Reset Identity Awareness Tables

This will clear out all the Identity Awareness tables:

```
fw tab -t pdp_sessions -t pdp_super_sessions -t pdp_encryption_keys -t pdp_timers -t pdp_expired_timers -t pdp_ip -t pdp_net_reg -t pdp_net_db -t pdp_cluster_stat -t pep_pdp_db -t pep_networks_to_pdp_db -t pep_net_reg -t pep_reported_network_masks_db -t pep_port_range_db -t pep_async_id_calls -t pep_client_db -t pep_identity_index -t pep_revoked_key_clients -t pep_src_mapping_db -t pep_log_completion -x -y
```

and it will fill up really fast, ½ second. So you can watch it with ‘pdp monitor all | wc’

**PDP Process**

PDP process is responsible for gathering security information about a person/system. With ANY packet that passes through an IA firewall, the PDP process attempts to identify the person/system the packet belongs to. If based on the SOURCE IP address the firewall is able to identity (not necessarily authenticate) a user associated with this address, the firewall will put the user into the PDP user table. At a minimum this association will be used to tag each log entry with the username.
The firewall PDP process does this starting with PEP. PEP gets a packet and looks at the IP address. PEP looks in its own internal table to see if there was an existing association. If not it asks PDP if it knows about the packet. If not, then PDP asks ADLOG if there were any WMI events with this IP.

If ADLOG has WMI events, the PDP asks the LDAP server if it any more information about the user. If the user exists in AD, then this user/IP/MACHINE are somewhat trusted under your AD environment. **BUT THEY ARE STILL! Not authenticated**, just one more level of authentication, enough to put a tag on the log entry.


Firewalls can implicitly/explicitly authenticate users/systems if the firewall policy requires the firewall to do this..
Before a user is allowed to pass through a firewall with rules like above the firewall must determine:

1. Identity
2. Verify identity (authentication)
3. Groups belong to (authorization)
4. System user is currently using (IP address)
5. Permitted time period
6. Application

The first step is identification/authentication. One way this is done with Captive Portal. If you configure a firewall to use a browser to authenticate, you have to set it up something like this.
This will force users to use a browser to authenticate to the firewall. Captive Portal is a web page the forces users to log in before they can use any other application. The user can try to pass through the firewall to a protected web server, and Captive Portal will intercept the request and bring you to this page. This web page must stay up during the duration of the application use.

NOTE: You don’t NEED to have Active Directory Query (WMI) enabled if you plan on using Captive Portal to authenticate. Captive Portal web page is the logon/logoff indicator. Might also save on slow WAN links. However, your logs might be somewhat barren of user-to-IP address mappings without it.
Here is the rulebase setup that will direct users to captive portal

**DANGER:** If a user fails Rule #12, rule processing will continue and the user may hit another rule. You MAY?? Want to cut the user off with a drop rule right away.

In the background the PDP process is responsible for collecting this information and storing it.
Here you can see the PDP process attached to the LDAP server via port 389.

The PDP process interacts with the LDAP/AD server by:

1. **LDAP search request**: Is this user in our forest
2. **LDAP response**: Get user info, user group membership
3. **LDAP bind request**: with user/password

4. **LDAP response**: YES/NO user is authenticated in the domain!!

5. Store the results in the PDP database
WARNING!!!!::: LDAP bind request: Test user identity/password against the AD server or else passwords will go in clear (make sure this is encrypted!)

TIMERS
Another IA crux are the timers.

If you look at the timers you will see that the Captive Portal user has 5 minutes to live.
This is where the 5 minutes comes from.

5 minutes and log them off

12 hours from WMI events
Here is another view of the timers.

Note: NO wmi koopalve events

5 minute timeout
There is one more way to set the timers. “Log out users when....” Option will override the 12 hour default. This option puts some Java magic in the browser so that every 5 minutes the firewall ‘pings’ the browser to see if its still alive. If someone shutdown their computer without gracefully logging off, then the firewall will force log them off and deletes entries from the pdp table.
If the user tries to close the captive portal browser, then the Java magic will force a logoff.

GROUPS – The Crux

Another crux is the group associations. It took us 1 year of working with CP developers to nail this all down. I also know there are plenty of SK’s on this topic because AD is funky how it reports group membership, nested groups. Universal Groups vs Domain groups, AUs not reporting group membership correctly if you have multiple AUs per domain….blah blah blah. Mostly an AD issue, not CP. Just wish we had more debugging tools.

Thank god, IA is easy to deploy and scalable. (more blue pills).

Once again PDP monitor shows us RELEVANT AD groups (not all the groups) that CP cares about WRT the user.
The only other way I know of debugging this is by tcpdump on port 389 during the LDAP SEARCH/RESPONSE.
Of course there is always the .elg files but you’ll be talking to CP support or a developer to get anything interesting in them.

ASIDE: I love CP support but debugging IA will need an especially savvy support person. Many support people just don't know AD so it’s a tough one. Go easy.

Now if you really want to geek out and look at all the underlying tables that support IA

```
fw tab -t pdp_sessions -t pdp_super_sessions -t pdp_encryption_keys -t pdp_timers -t pdp_expired_timers -t pdp_ip -t pdp_net_reg -t pdp_net_db -t pdp_cluster_stat -t pep_pdp_db -t pep_networks_to_pdp_db -t pep_net_reg -t pep_reported_network_masks_db -t pep_port_range_db -t pep_async_id_calls -t pep_client_db -t pep_identity_index -t pep_revoked_key_clients -t pep_src_mapping_db -t pep_log_completion
```

**PDP Debug**

This is the crux of IA, just getting the PDP tables to populate. You probably don’t have access to AD debug and detailed logging so you will just get CP log info and then you will have to call your AD buddies and hope they pick up the phone. This is why if you are an AD domain admin IA is so much easier to debug. If not, you will have to rely on blue pills and patience.

(This is section will be my next boring blog because there is so much info to cover, but here are some tidbits to get you started).

Remember that if a user shows up in PDP, that only means the firewall has seen a packet and has tried to tie user information to it. Nothing else. No authentication, etc.

“pdp monitor all” is the start of all your debugging. This lists all the packets the firewall has attempted to identify. You will get a list of user-to-ip mappings of both firewall AUTHENTICATED users and users just passing through minding their own business.
This user AUTHENTICATED by Captive Portal

Captive Portal Identify and AUTHENTICATE

This user was IDENTIFIED by WMI and AD. Not authenticated. Just passing thru no Captive Portal rules hit.

AD Trust to IDENTIFY
I already talked about PEP so will wrap it up a bit.
PEP is responsible for ENFORCEment of the rulebase which allows/permits AD users and AD machines (not IP addresses) through a firewall. So just like PDP, PEP keeps a table/cache of IP/Users that it has seen pass through the firewall.

This table replicates the pdp table. This table does NOT say “these IP addresses are allowed through the firewall”. It only says “I am trying to identify packets I have seen”.

Why have this table if PDP already has one?? Well not all sites have AD domain controllers. So instead of flooding a remote site with tons of WMI traffic, just share a PDP table between firewalls.
And here is how you do that:

Check Point Gateway - Corporate-Identity-Awareness

Identity Awareness

- Browser-Based Authentication
  - The portal is accessible only through internal interfaces.
  - Portal URL: https://identity.myvcorp.local/connect
- Active Directory Query
  - Learn more...
- Identity Agents
- Terminal Servers
- Remote Access

Identity Sharing

- Get identities from other gateways:
  - Server Name
  - All sharing gateways
- Share local identities with other gateways

Remote non-AD site

AD site
PEP/Portal Sharing

There is this cool feature you can use to:

1. Centralize your portals
2. Limit the number of portals you have to configure and support
3. Users only have to remember 1 portal instead of 100
4. Centralize the portal, but if it breaks you can still authenticate at the local portal

So it looks something like this:

1) User authenticates at central Captive Portal
2) User can then have access to ANY of the sites whose firewalls share that user database on fw-1:

How is this all done?? Well, you first have to allow the fw-1 system to share identities with the remote sites:
And on the remote sites you have to point to the fw-1: corporate identity awareness portal:

Let’s look what happened underneath the covers.

PEP and PDP setup connections with each other. PEP needs to share the user table that PDP is holding on the fw-1: Corporate IA Portal
These are the ports that are being used to do that. You can see there are the main ports and then some random ports it spawns off.
So now comes the fun part. Not sure how your network setup is, but I saw connections come from all interfaces to all interfaces. I didn’t debug but I think it depends on which interface
Captive Portal is running. Anyways the result was a ton of spoofing drops. Took me a while to figure out what was going on.

In addition, if you have firewalls between the fw-1 and the remotes, they you have to open these ports so they can talk IA.

So to fix this whole mess, I had to put the firewalls into spoof groups on all interfaces.
Install policy on both fw1 Corporate IA Portal and fw2 remote.

After I did this CP actually has been working awesome so far. Remember this is with multiple AUs in a single AD domain and a single MDS domain - which is what CP strongly warns you against.
PEP Debug

Not much to say here. If you debug pdp, then pep should start to work. Only one command you need:

**pep show user all**

<table>
<thead>
<tr>
<th>ID (PDP; UID)</th>
<th>Username@Machine</th>
<th>CID (IP, PacketID)</th>
<th>PT</th>
</tr>
</thead>
<tbody>
<tr>
<td>127.0.0.1:00000000; 5242dc9</td>
<td>user1:user2:user3</td>
<td>1.1.1.1</td>
<td>0</td>
</tr>
<tr>
<td>127.0.0.1:00000000; b4344dc9</td>
<td>user4</td>
<td>2.2.2.2</td>
<td>0</td>
</tr>
<tr>
<td>127.0.0.1:00000000; 959ec7c9</td>
<td>user5</td>
<td>2.2.2.2</td>
<td>0</td>
</tr>
<tr>
<td>127.0.0.1:00000000; 35af4930</td>
<td>user6</td>
<td>2.2.2.2</td>
<td>0</td>
</tr>
<tr>
<td>127.0.0.1:00000000; 9946121a</td>
<td>user7</td>
<td>2.2.2.2</td>
<td>0</td>
</tr>
<tr>
<td>127.0.0.1:00000000; 5edb31c9</td>
<td>user8</td>
<td>2.2.2.2</td>
<td>0</td>
</tr>
<tr>
<td>127.0.0.1:00000000; 1af60b0</td>
<td>user9</td>
<td>2.2.2.2</td>
<td>0</td>
</tr>
</tbody>
</table>

And some .elg files

```
[Expert@....]# !cd
cd $FWDIR/log
[Expert@....]# !ls
ls pep*
pepd.elg  pepd_clsi.elg
[Expert@....]# ![image]
```

I'm sure there are more, just haven’t needed it to this point.

Summary:
SOLUTION

Traditionally, firewalls use IP addresses to monitor traffic and are unaware of the user and machine identities behind those IP addresses. Identity Awareness removes this notion of anonymity since it maps users and machine identities. This lets you enforce access and audit data based on identity.

Identity Awareness is an easy to deploy and scalable solution. It is applicable for both Active Directory and non-Active Directory based networks as well as for employees and guest users. It is currently available on the Firewall blade and Application Control blade and will operate with other blades in the future.

Special Thanks to The Gene Machine (Gene Berger) who spent many hours explaining the details of IA to me and showing me the shortcuts. So patient. Much luck in your future as a CheckPoint SE!!