ClearPass and iboss Integration
ClearPass and iboss integration
TechNote

Change Log

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Modified By</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>Apr 2015</td>
<td>Danny Jump</td>
<td>Early Draft Versions</td>
</tr>
<tr>
<td>0.2</td>
<td>May 2015</td>
<td>iboss</td>
<td>Reviewed and amended by iboss</td>
</tr>
<tr>
<td>1.0</td>
<td>May 2015</td>
<td>Danny Jump</td>
<td>First Published Version</td>
</tr>
<tr>
<td>1.1</td>
<td>Feb 2016</td>
<td>Danny Jump</td>
<td>Minor Updates</td>
</tr>
<tr>
<td>2.0</td>
<td>Mar 2017</td>
<td>Danny Jump</td>
<td>Minor Updates</td>
</tr>
</tbody>
</table>

Copyright

© Copyright 2017 Hewlett Packard Enterprise Development LP.

Open Source Code

This product includes code licensed under the GNU General Public License, the GNU Lesser General Public License, and/or certain other open source licenses. A complete machine-readable copy of the source code corresponding to such code is available upon request. This offer is valid to anyone in receipt of this information and shall expire three years following the date of the final distribution of this product version by Hewlett- Packard Company. To obtain such source code, send a check or money order in the amount of US $10.00 to:

Hewlett-Packard Company
Attn: General Counsel
3000 Hanover Street
Palo Alto, CA 94304
USA

Please specify the product and version for which you are requesting source code. You may also request a copy of this source code free of charge at HPE-Aruba-gplquery@hpe.com.
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction and Overview</td>
<td>5</td>
</tr>
<tr>
<td>Software Requirements</td>
<td>5</td>
</tr>
<tr>
<td>ClearPass Installation and Deployment Guide</td>
<td>5</td>
</tr>
<tr>
<td>ClearPass Policy Manager</td>
<td>6</td>
</tr>
<tr>
<td>Overview of Policy Manager integration with iboss</td>
<td>6</td>
</tr>
<tr>
<td>Enabling Profiling &amp; Insight</td>
<td>6</td>
</tr>
<tr>
<td>Configuration of Context Server</td>
<td>7</td>
</tr>
<tr>
<td>Configuration of Context Server Actions</td>
<td>8</td>
</tr>
<tr>
<td>Logout User to iboss Security Platform</td>
<td>10</td>
</tr>
<tr>
<td>Configuration of Enforcement Profiles/Policies</td>
<td>12</td>
</tr>
<tr>
<td>iboss Configuration</td>
<td>15</td>
</tr>
<tr>
<td>Overview of iboss</td>
<td>15</td>
</tr>
<tr>
<td>iboss Configuration</td>
<td>15</td>
</tr>
<tr>
<td>Defining source subnets in iboss</td>
<td>16</td>
</tr>
<tr>
<td>Defining CleaPass Nodes in iboss</td>
<td>17</td>
</tr>
<tr>
<td>Defining Groups/Roles in iboss</td>
<td>18</td>
</tr>
<tr>
<td>iboss Enforcement</td>
<td>19</td>
</tr>
<tr>
<td>iboss Troubleshooting</td>
<td>20</td>
</tr>
<tr>
<td>Appendix A – iboss API Detail</td>
<td>22</td>
</tr>
</tbody>
</table>
Figures

Figure 1: Enabling Profiling .......................................................... 6
Figure 2: Enabling Insight .............................................................. 7
Figure 3: Adding iboss Context Server .......................................... 7
Figure 4: Endpoint Context Server - Action TAB .......................... 8
Figure 5: Copy of the Context Server Action URL ....................... 8
Figure 6: Endpoint Context Server - Header TAB ......................... 9
Figure 7: Endpoint Context Server - Content TAB ....................... 9
Figure 8: Logout user from iboss - Action TAB ........................... 10
Figure 9: Logout user from iboss - Header TAB .......................... 10
Figure 10: Logout user from iboss - Content TAB ....................... 11
Figure 11: Enforcement Profile, adding a context-server-action for a Group/Role ................................................. 12
Figure 12: Enforcement Profile matching on the service role mapping ................................................................. 13
Figure 13: Role mapping for users .................................................. 14
Figure 14: Setting Security PSK and listening port ...................... 15
Figure 15: Defining the iboss local-subnets ................................. 16
Figure 16: Adding the iboss subnets and setting the Default Policy ................................................................. 16
Figure 17: Adding SOURCE ClearPass nodes to iboss .................... 17
Figure 18: Defining iboss GROUPS to match ClearPass roles .......... 18
Figure 19: Using iboss to enforce user/group restrictions ............ 19
Figure 20: Looking at the 'Last Communication from ClearPass to iboss ................................................................. 20
Figure 21: Analyzing inbound request from different ClearPass servers ................................................................. 21
**Introduction and Overview**

The solution below provides the ability for ClearPass Policy Manager to share network logon credentials and device level context of an authenticated user (IP address, Hostname, User-Role/Group) with the iboss security platform. Additional context can be shared, and it is documented fully in Appendix A.

**Software Requirements**

The minimum software version required for ClearPass is 6.5.2. At the time of writing, ClearPass 6.6.4 is the latest available and recommended release. Any subsequent ClearPass software release will support this integration. ClearPass runs on either hardware appliances with pre-installed software, or as a Virtual Machine under the following hypervisors. Hypervisors that run on a client computer such as VMware Player are not supported.

- VMware ESXi 5.0, 5.1, 5.5, 6.0, 6.5 or higher
- Microsoft Hyper-V Server 2012 or 2016 R2
- Hyper-V on Microsoft Windows Server 2012 or 2016 R2
- KVM on CentOS 6.6, 6.7, or 6.8.

**ClearPass Installation and Deployment Guide**

This document assumes your ClearPass environment is already configured and operational. If you require assistance with basic deployment, refer to the following deployment guide:

ClearPass Policy Manager

Overview of Policy Manager integration with iboss

Support for integration with iboss WebSecurity is delivered using the ClearPass Exchange Framework. The ClearPass Exchange framework expands the reach of the business rules that can be defined and enforced as part of an Access Management solution by introducing the concept of HTTP enforcement. By complementing the existing network policy enforcement and protocol support such as RADIUS, HTTP enforcement allows ClearPass to interact with an ever-increasing list of API-enabled 3rd party solutions.

The use-case below takes user authentications, matching users to AD group membership, and assigns a ClearPass role based upon the AD group membership. Then this role is used as the basis to match to a group inside the iboss security gateway to restrict/permit user access.

Before jumping into the detailed configuration covering the interaction between ClearPass and iboss we need to ensure that a few core basic features are enabled on ClearPass.

Enabling Profiling & Insight

It important to ensure that Profiling and Insight are enabled. Check this under Administration -> Server Manager -> Server Configuration -> [chose you server]

Figure 1: Enabling Profiling

![Figure 1: Enabling Profiling](image-url)
In the same location check Insight is enabled

**Figure 2: Enabling Insight**

![Image of Server Manager](screen1.png)

Once these are enabled ensure that the NAS/NAD that are forwarding network authentication to ClearPass are also forwarding RADIUS Accounting data. Accounting data is used to correlate the IP address of the endpoint. Without the endpoint IP address, we are unable to update the iboss server.

**Configuration of Context Server**

Several components need to be configured for the integration and sharing of user/device context. First, the iboss system needs to be defined within ClearPass. Go to Administration -> External Servers -> Endpoint External Servers -> Add [Generic HTTP]

**Figure 3: Adding iboss Context Server**

![Image of Endpoint Context Server](screen2.png)

Ensure you add the endpoint context server as type Generic HTTP, iboss now supports HTTPS as well as HTTP. Finally, notice we do not add any username/password credentials.
**Configuration of Context Server Actions**

Next, we need to define the actions and the formatting of the data that is sent to the iboss platform. This is completed under **Administration -> Dictionaries -> Context Server Actions -> Add a Generic HTTP action**. The example below shows how to configure the actions to send Username, IP address, hostname, domain-name and a Group. In our example the Group comes out of TIPS role mapping which is derived from AD memberOf groups as discussed above. You can amend as appropriate to fit the needs of the customer. There may be other context outside of AD that you want to use to assign the role.

Due to a restriction that is beyond the scope of this document, currently you will need to create a context-server-action for each group-mapping/role you intend to use.

**Login User to iboss Security Platform**

Below is the example of creating an **Action** for a mapping to a group/role ‘PLM’. Notice the group via the &g attribute is hardcoded to PLM. You need to duplicate this configuration and modify the &g as appropriate.

**Figure 4: Endpoint Context Server - Action TAB**

Pay special attention to the **URL** section. A full example is below. Two attributes need to be set. One is the &key PSK that is obtained from the iboss administrator (this is covered in the iboss section later in the document). The second is the &g=XXX, this is the group/role/label attribute used to segment users.

If you choose HTTPS when creating the context-server, then the port number above needs to be 8016.

**Figure 5: Copy of the Context Server Action URL**

```
:8015/nacAgent?action=logon&user=\%{name}&wn=\%{machine}&ip=\%{ip}&udomain=\%{domain}&g=PLM&key=XS8123ABC
```
As mentioned above its suggested you build separate context-server-actions and statically define the group/role in this way for each group/role we intend to pass to iboss. If we extract memberOf from Active-Directory, multiple roles might be returned and sending multiple groups to iboss would now allow a match.

The **Header** attributes are here for reference, note the Content-Type below.

*Figure 6: Endpoint Context Server - Header TAB*

Within the **Content** header, you need to define the attributes shown below. These are referenced within the URL API on the **Action** Tab.

*Figure 7: Endpoint Context Server - Content TAB*

A full copy of the **Content** tab attributes is shown below. This section can be copy/passed as required.

```json
{"user":"%{name}","ip-address":"%{ip}","machine":"%{machine}","domain":"%{domain}"}
```
Logout User to iboss Security Platform

In addition to the login-action, create a context-server-action that will logout users from the iboss platform. This is much simpler than the login/add users. Only a single logout action needs to be configured. Only three parameters are configured on the Action tab. The user, the IP address and the PSK of the iboss node.

**Figure 8:** Logout user from iboss - Action TAB

![Image](image.png)

Below is a copy of the URL string that can be copy/pasted. If you choose HTTPS when creating the context-server, then the port number needs to be configured as 8016.

```
:8015/nacAgent?action=logoff&user=%{name}&ip=%{ip}&key=XS8123ABC
```

Change the \&key parameter to match your own system.

A copy of the **Header** is shown below; this is the same as in the login/add user's definition.

**Figure 9:** Logout user from iboss - Header TAB

![Image](image.png)
In the same way as in the login/add user, we must define the attributes we will use in the **Action** Tab, here also within in the **Content** Tab.

**Figure 10:** Logout user from iboss - Content TAB

![Endpoint Context Server Details](image)

A copy of the Content that can be copy/pasted is below.

```
{"user":"%{name}","ip-address":"%{ip}"}
```

Continuing from the above configuration, the remaining configuration is standard ClearPass Policy Manager profiles/policies/service. An example is provided later in the document for the inexperienced user.
Configuration of Enforcement Profiles/Policies

As mentioned before, the configuration of the remaining Policy Manager items is traditional services, enforcement profiles and enforcement policies. Below is an overview of the enforcement profiles/policy configuration, however it is assumed you have an understanding of the service policy components and are capable of adding a new profile to an existing policy.

Configure one Enforcement Profile per role/group that you want to enforce with iboss. Go to Configuration -> Enforcement -> Profiles -> add a new profile of type ‘Session Notification Enforcement’ then under ‘Session-Notify’ add the items shown below:

- Server-Type set as ‘Generic HTTP’
- Server IP, i.e the IP address of the iboss Security node.
- Login/Logout set as your context-server-actions

In the example below, the Login Action profile is hardcoded specifically to send ‘TME’ as the group attribute to the iboss. You would in essence replicate this Profile to reference multiple context-server-actions, each action having a defined role/group configured.

Figure 11: Enforcement Profile, adding a context-server-action for a Group/Role

The Logout is triggered when ClearPass receives an accounting stop for a session.

Next configure or modify an existing Enforcement Policy. Either add the profile from the previous step to existing conditions rules or build a new one as appropriate, you’ll need to replicate the above for each role/group you want to use.
Figure 12: Enforcement Profile matching on the service role mapping

Here is another example where we have added the iboss login to an existing policy.
Here as an example of additional configuration, we have taken the AD memberOf data and built a role-mapping rule which checks for the occurrence of a match through the CONTAINS condition and then assigns a role to the user.

**Figure 13:** Role mapping for users
iboss Configuration

Overview of iboss

iboss describes itself “iboss Cybersecurity protects today's borderless networks against malware, advanced threats and data loss with innovative Web Security, Mobile Security and FireSphere™ advanced APT defense. Backed by patented technology, iboss' stream-based approach delivers unparalleled visibility across all inbound/outbound data channels and port-evasive applications”. In the integration with ClearPass Policy Manager, ClearPass becomes the Network Authentication Policy guardian, we can enhance iboss, by providing visibility of the wired/wireless users and providing context to iboss of how to enforce policy for these users. By sharing context we have about the user, be that an Enterprise or a Guest user, the integration allows the iboss platform to seamlessly and transparently enforce accurate user policies.

iboss Configuration

The first item required from the iboss platform is the PSK we configure in the base URL in the context-server-action. The iboss system generates a default one or you can create your own. From the iboss UI navigate to [HOME] -> ‘User Single Sign-On’ -> AD Plugin then Global Settings. Below the Security PSK is highlighted and the default port used to ‘listen’ for the incoming HTTP/s commands from ClearPass.

Note the highlighted point below about changing the listening port from its default settings.

**Figure 14: Setting Security PSK and listening port**
Defining source subnets in iboss

The next step is to define in the iboss platform a list of the subnets the users will be authenticating from, you must identify the source subnets in iboss. Define the subnets here [HOME] -> Network -> ‘Local Subnets’ -> ‘+ New Local Subnet/IP Range’

**Figure 15: Defining the iboss local-subnets**

![Local Subnets/IP Ranges](image)

When adding the subnet definitions ensure that you change the Default Policy as shown below to be ‘No, Bypass Filtering Rules for this Subnet’

**Figure 16: Adding the iboss subnets and setting the Default Policy**

![Add Local Subnet/IP Range](image)
**Defining CleaPass Nodes in iboss**

Next define the IP address of the source ClearPass system that will be sending data to the iboss system. If this is not configured or does not match, then the iboss Platform will drop the incoming request. Note that if you’re sending data from a ClearPass node that is located behind a firewall/nat-device then the SRC IP address that the iboss node will try to match on will not be the actual IP address of the physical system but the public IP address of the firewall. To define the ClearPass nodes go to [HOME] -> ‘User Single Sign-On’ -> AD Plugin -> Registered AD Servers / NAC Agents [and then add a new node] as shown below.

*Figure 17: Adding SOURCE ClearPass nodes to iboss*
Defining Groups/Roles in iboss

The final step in the configuration is the creation of the iboss Groups. In essence these are the Groups that will be used to match and then enforce a policy restriction. Think of these iboss Groups as matching ClearPass roles. So, as an example, we might process an authentication and refer to the users AD group membership to derive their role. This role we send to iboss to allow it to match users based upon this role/grouping label.

Find the group definitions configuration [HOME] -> ‘Users, Groups & Devices’ -> ‘Users & Devices’

**Figure 18: Defining iboss GROUPS to match ClearPass roles**

In the above on the left-hand-side, several of the default groups were renamed to align with the groups/roles we want to match users against. Above we renamed groups 2-6 to be something more meaningful and as such the role ClearPass sends will match the group name above. Note that the endpoint details we send to iboss appear under the ‘**Dynamic Devices**’ tab as highlighted above.
**iboss Enforcement**

Now that user and device context and importantly a role is being received by iboss this can then drive the power of the iboss security platform to enforce/restrict user’s ability and actions. It’s beyond the scope of this document to cover all the enforcement examples available but as an example, in iboss control can be applied to a user’s ability to use Different Chat applications, File extensions, Social Media Applications P2P traffic, Bandwidth-shaping, etc.

Below we show one of the categories and its ability to select policy control per Group based upon the ClearPass Role.

**Figure 19: Using iboss to enforce user/group restrictions**

The above, shows an overview of the type of enforcement possible. The policy enforcement within iboss allows for an extensive set of controls.
iboss Troubleshooting

If communication problems exist there are a couple of places you can go to see if we are communicating with the iboss Security platform. From [HOME] > ‘User Single Sign-On’ > AD Plugin then expand ‘Last Communication Info’.... Set a filter based upon the expected user-name as can be seen below, information regarding the last communication, including date and time and details about the HTTP request we sent should show up in the Request Info window.

*Figure 20: Looking at the 'Last Communication from ClearPass to iboss'*

Additionally under the ‘Registered AD Servers /NAC Agents’ as below, we can see the total number of Requests, broken down by successful and unsuccessful, then per-server statistics showing the number of inbound requests and the number of successful and unsuccessful requests.
Figure 21: Analyzing inbound request from different ClearPass servers
Appendix A – iboss API Detail

The table below describes all the iboss NAC API request parameters in additional detail.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>user</td>
<td>String</td>
<td>Yes</td>
<td>The username to associate with an IP address. Not required if the request is to request information about an IP Address.</td>
</tr>
<tr>
<td>cn</td>
<td>String</td>
<td>No</td>
<td>The full name of the user.</td>
</tr>
<tr>
<td>g</td>
<td>String</td>
<td>No</td>
<td>The groups/OUs that the user belongs to. These are comma delimited and will be matched one by one to the group names configured in iboss.</td>
</tr>
<tr>
<td>ip</td>
<td>String, comma delimited</td>
<td>Yes</td>
<td>The IP Address to associate a user with or the IP Address of which the NAC wants information about.</td>
</tr>
<tr>
<td>dc</td>
<td>String</td>
<td>No</td>
<td>The name of the NAC. Should match the name registered in the NAC registration page, but can differ.</td>
</tr>
<tr>
<td>key</td>
<td>String</td>
<td>Yes</td>
<td>The security key for the API request. This must match the key in the NAC registration page.</td>
</tr>
<tr>
<td>loc</td>
<td>String</td>
<td>No</td>
<td>The location of the user. Can be any friendly string.</td>
</tr>
<tr>
<td>wn</td>
<td>String</td>
<td>No</td>
<td>The user’s computer name</td>
</tr>
</tbody>
</table>
| action    | String | Yes      | Can have the values logon, logout, or clientInfo.  

- **logon** = This is a user logon event and the NAC wishes to associate a user and policy to an IP Address.  
- **logoff** = The user has logged off an IP Address and the NAC wishes the iboss to remove the association between the user and IP Address within iboss. The policy in iboss will return to the default policy for that IP Address.  
- **clientInfo** = The NAC wishes to obtain user information about an IP Address and determine which user is registered to an IP Address in iboss. Can be used for synchronization. The iboss returns the associated user, policy group number, name as well as other information. |
| udomain   | String | No       | The domain to which the user belongs. Can be blank. |
| defaultGroup | Integer | No | Allows the NAC to specify what the default group should be if none of the group names submitted match iboss groups or an LDAP group match failure has occurred. This overrides the subnet default group and the selected NAC default group on the NAC registration page. The default group should be submitted as a number from 1 to the max number of groups available on the iboss model. |
| forceGroup | Integer | No | This allows the NAC to specify which group policy should be used for a user. This overrides group/OU matching and allows the NAC to force a group for a user overriding any other criteria for group matching. This should be submitted as a number from 1 to the max number of groups available on the iboss model. |