CYBERARK-PROOFPOINT INTEGRATION

Name of Company: Proofpoint

Website: www.proofpoint.com

Name of Product: Threat Response

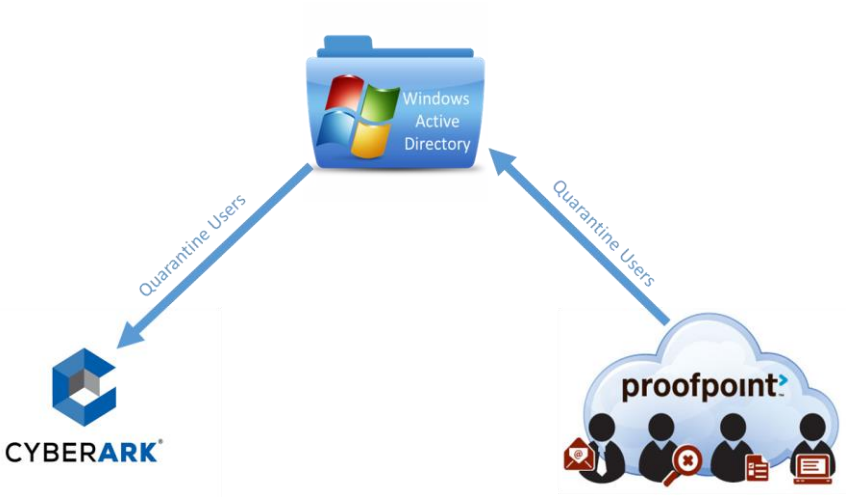
Date: June 2016

KEY BENEFITS

Armed with Proofpoint Threat Response insight, a Cyber Defender can quarantine a potentially compromised user account. When that quarantined user tries to utilize privileged credentials, the CyberArk Privileged Account Security Solution can immediately implement higher security controls around that user's privileged access rights and capabilities on critical assets.

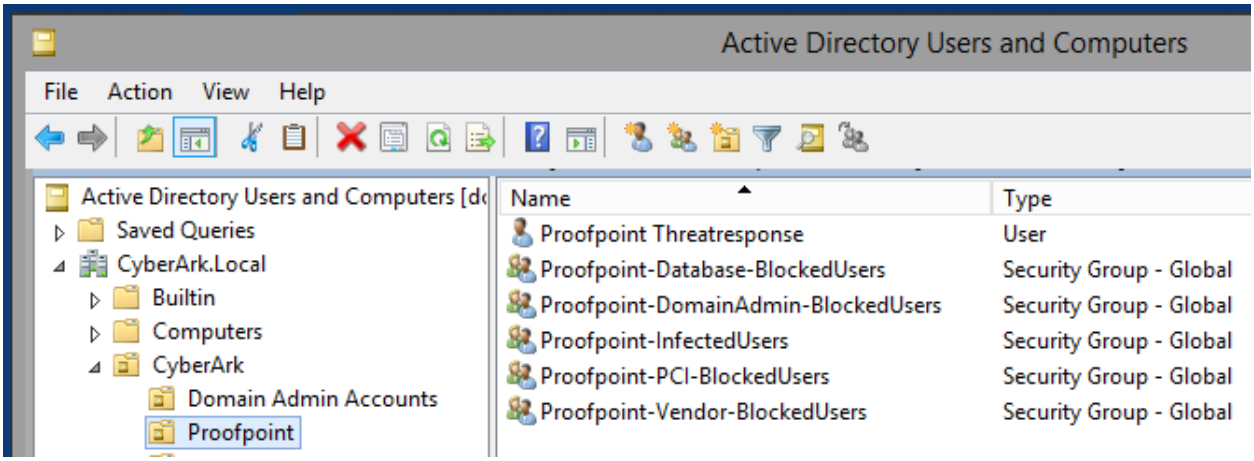
This integration between Proofpoint Threat Response and the CyberArk Solution helps mitigate the risk of a potential breach by making it difficult for attackers to access an organization's IT infrastructure, disable security controls, steal confidential information, commit financial fraud and disrupt operations.

PRODUCT DIAGRAM & DESCRIPTION OF PRODUCT INTEGRATION



The integration between Proofpoint Threat Response and the CyberArk was developed around the CyberArk's PVWA Ticketing Integration framework, which provides a powerful user-exit capability, used in this particular case, to validate whether a user has being quarantined before granting access to a Privileged Account (Show, Copy, or Connect).

Because this integration is done through Active Directory, you must have CyberArk already configured with AD before proceeding. The PVWA relies on your AD Domains and on the quarantine groups, managed by Proofpoint:

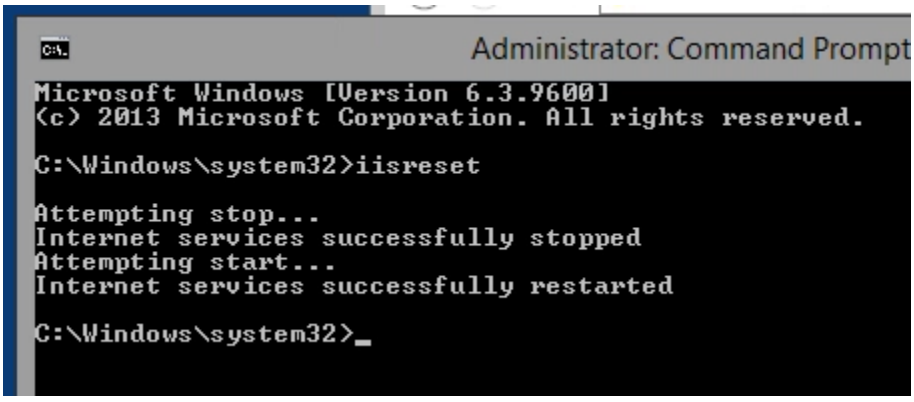


The code is delivered through a single DLL file (ProofpointValidator.dll); the interaction with Proofpoint is governed by a specific set of Ticketing Parameters described in the Configuration section of this document.

The integration defines a hierarchy: Domain -> Rule -> Condition. Where Domain and Rule are mandatory. These are defined in the Ticketing System parameters section on the PVWA Administration page. You can specify as many parameters as required. The parameters can be specified in different hierarchies and levels and are transferred to the integration module in an xml object when a user clicks on Show, Copy, or Connect. Based on the logic defined by these parameters, the integration module determines if the user will be denied access. See Appendix 1 for an example of the XML object.

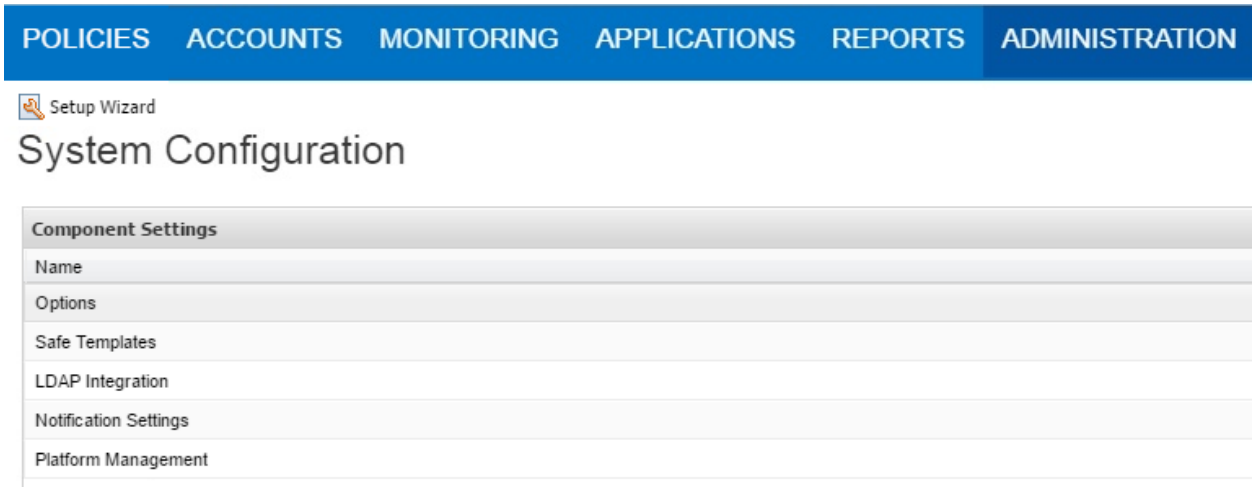
INSTALLATION

Copy **ProofpointValidator.dll** to **%inetpub%\wwwroot\PasswordVault\Bin** on the PVWA server, then restart IIS.

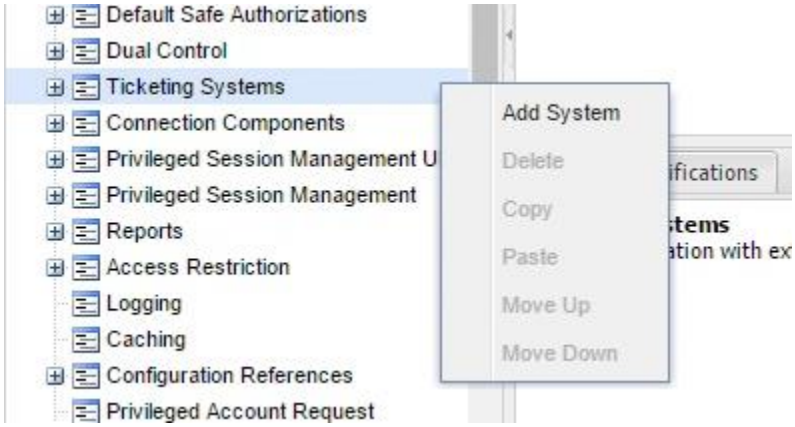


CONFIGURATION

Log-in to the PVWA as a vault administrator user, then go to **ADMINISTRATION** and click on **Options**:



Right click on **Ticketing Systems** and select **Add System**:

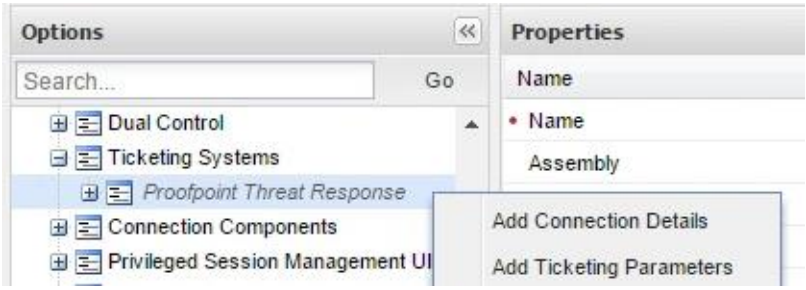


On the **Properties** section enter the following information:

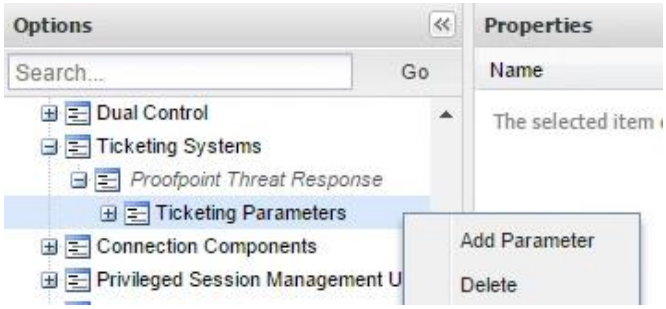
Name: You can enter any value.

Assembly: Corresponds to the name of the dll file provided in the package. You *must* enter **ProofpointValidator** as shown here.

After clicking **Apply**, Proofpoint will appear under **Ticketing Systems**. Right click on it and select **Add Ticketing Parameters**:

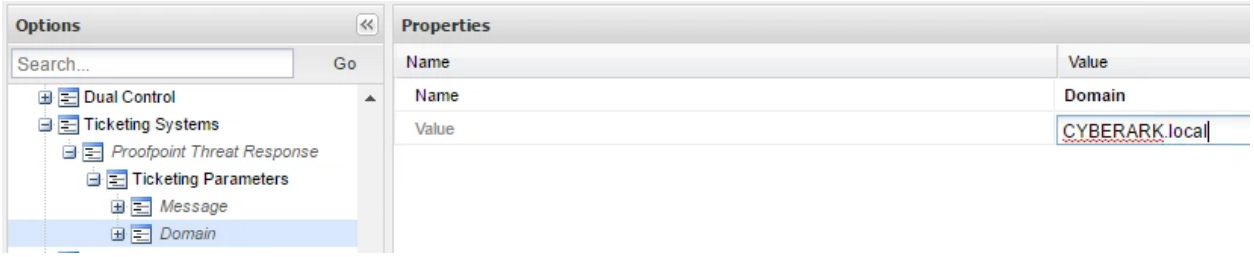


A new **Ticketing Parameters** section will appear under Proofpoint. Right click on it and select **Add Parameter**:

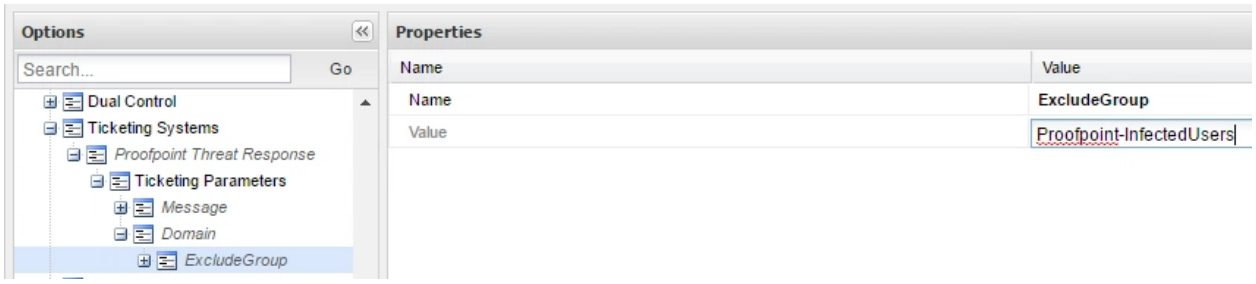


Name this Parameter **Message** and enter a generic message you want the PVWA to display when denying access to a quarantined user. This message will apply to all Domains in your AD.

You will be able to define additional messages, which will follow the generic message, providing specific details, in the following subsections. Repeat the process to add a new parameter called **Domain**. The value of this parameter *must* be one of your Active Directory domains:

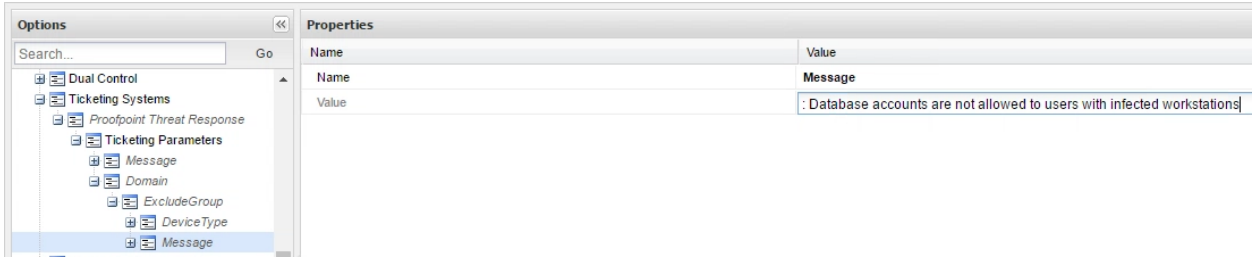


This integration release supports only one type of rule, **ExcludeGroup**, which is one of the AD groups maintained by Proofpoint containing quarantined users. However, you can add multiple rules per domain. Right Click on Domain to Add a Rule:



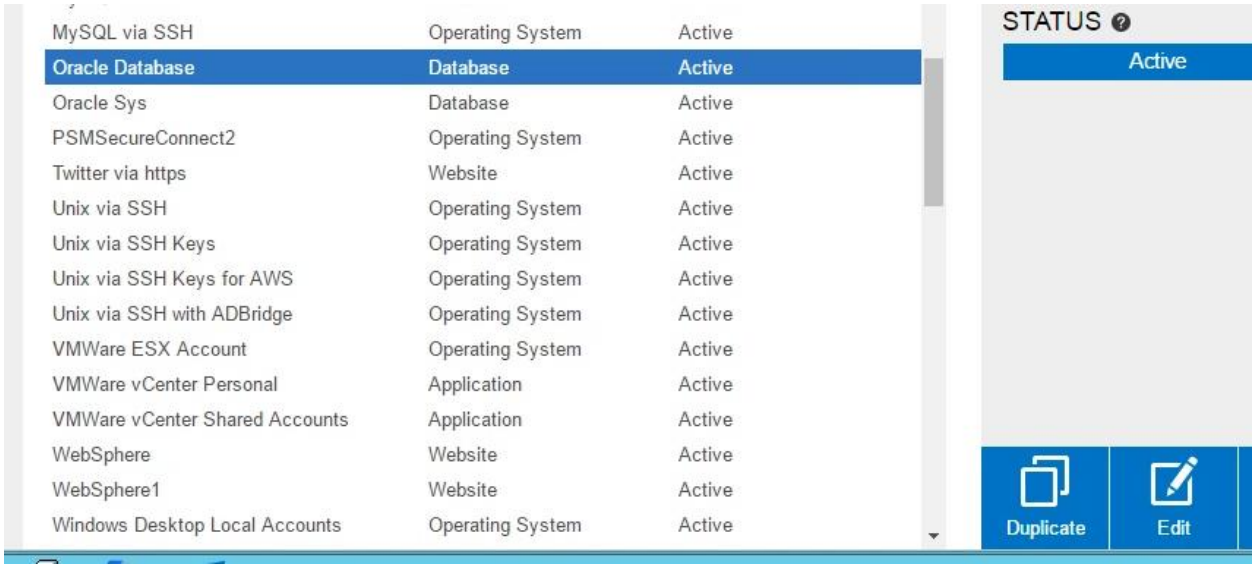
A rule by itself is sufficient to deny access to a quarantined user. You can also add conditions to the rule to further refine what a quarantined user will be allowed to access, based on properties of the target account (refer to Appendix 2 for more details). For example, let's say we want to limit the quarantine to Database devices only while allowing him/her access to all other device types:

You should also define a specific message (optional) to the Rule (ExcludeGroup) oe Condition (DeviceType), which will be concatenated to the generic message, to add more details. You should have one Message parameter per rule, or per condition.

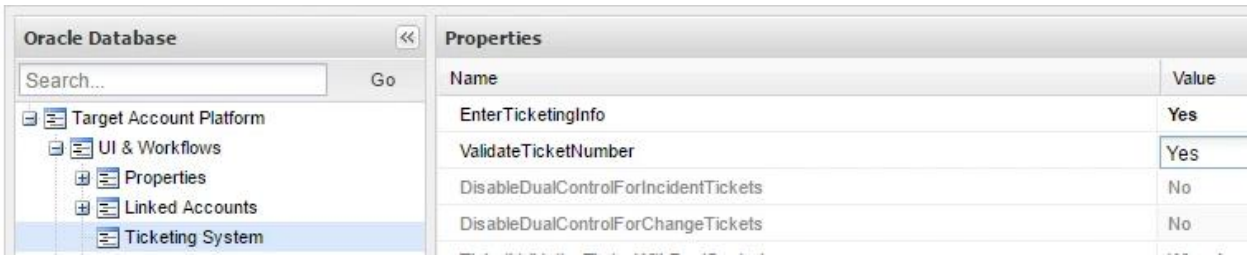


ENABLING THE INTEGRATION FOR THE RELEVANT PLATFORMS

Enable Proofpoint Integration (Ticketing System) for platforms where you want to deny quarantined users. Log-in to the PVWA as a vault administrator user, go to **ADMINISTRATION** and click on **Platform Management**. Select your platform and click on **Edit**:



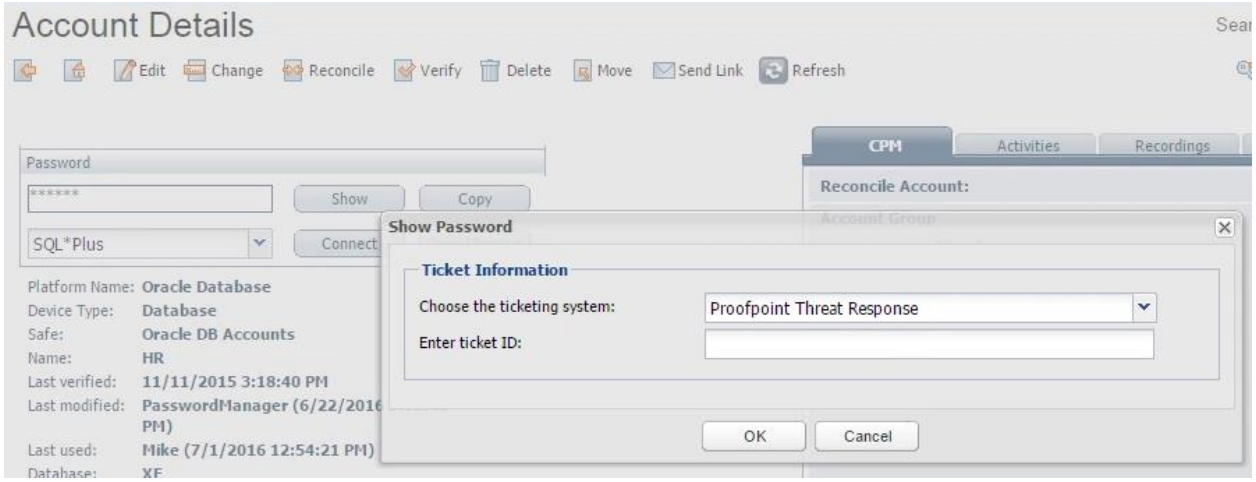
If you do not already have an active Ticketing System, then expand **UI & Workflows**, click on **Ticketing System**, and set both **EnterTicketinginfo** and **ValidateTicketNumber** to **Yes**. Then click **OK**:



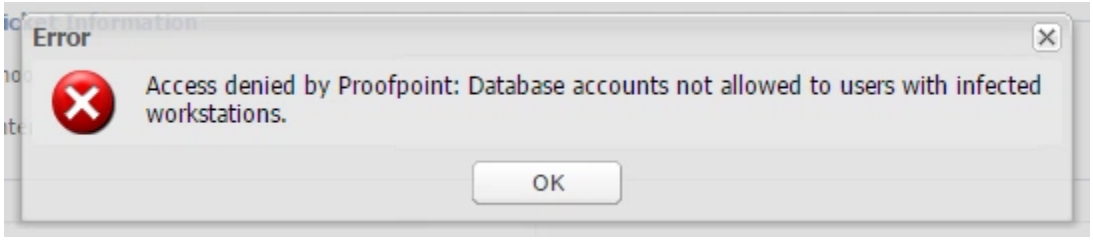
Note: If you have multiple ticketing systems you can enable only some of them for a specific platform, please refer to section "*Integrating with Ticketing Systems*" of the "*Privileged Account Security Implementation Guide*" documentation.

TESTING

To test, login as a quarantined user, select an account belonging to the **DeviceType** you entered, and click on **Show**. You should see the Ticketing System dialog:



Click on OK and you'll see the error message denying access to password:



APPENDIX 1: XML EXAMPLE

```
<TicketingParameters>
	<Parameter Name="Message" Value="Access denied by Proofpoint" />
	<Parameter Name="Domain" Value="CYBERARK.local">
		<Parameter Name="Message" Value=" [CYBERARK.local] " />
		<Parameter Name="NotMemberOfGroup" Value="Proofpoint-InfectedUsers">
			<Parameter Name="Message" Value=": Quaratined Infected User (Proofpoint-InfectedUsers)" />
		</Parameter>
		<Parameter Name="NotMemberOfGroup" Value="Proofpoint-PCI-BlockedUsers">
			<Parameter Name="Message" Value=": Quarantined PCI Access (Proofpoint-PCI-BlockedUsers)" />
			<Parameter Name="REGEX:PolicyId" Value="(PCI)">
				<Parameter Name="Message" Value=" - Platform matched PCI" />
			</Parameter>
			<Parameter Name="REGEX:SafeName" Value="(PCI)">
				<Parameter Name="Message" Value="- Safe Matched PCI" />
			</Parameter>
		</Parameter>
		<Parameter Name="NotMemberOfGroup" Value="Proofpoint-Database-BlockedUsers">
			<Parameter Name="Message" Value=": Quarantined Database Access (Proofpoint-Database-
BlockedUsers)" />
			<Parameter Name="DeviceType" Value="Database">
				<Parameter Name="Message" Value=" - Target Device is Database" />
			</Parameter>
		</Parameter>
		<Parameter Name="NotMemberOfGroup" Value="Proofpoint-DomainAdmin-BlockedUsers">
			<Parameter Name="REGEX:PolicyId" Value="Domain" />
			<Parameter Name="REGEX:SafeName" Value="(Domain)" />
		</Parameter>
		<Parameter Name="NotMemberOfGroup" Value="Proofpoint-Vendor-BlockedUsers">
			<Parameter Name="REGEX:UserName" Value="vendor" />
		</Parameter>
	</Parameter>
</TicketingParameters>
```

APPENDIX 2: SUPPORTED PROPERTIES FOR CONDITIONS

You can add conditions to rules to further refine what a quarantined user will be allowed to access, based on properties of the target account, or the requesting user.

Property	Description
SafeName	The Safe where the requested account is stored.
MachineAddress	The address of the machine specified on the account, if it exists.
TransparentMachine Address	The address of the machine used in a transparent connection, if it exists.
UserName	The name of the user specified on the requested account.
PolicyId	The name of the platform specified on the requested account.
RequestingUser	The name of the user requesting the account.

In addition to the properties listed above, all target account properties are available, such as: DeviceType, Port, Database, LogonDomain, etc.

Supporting exact match comparisons or regular expressions you can add logic like:

- Do not allow if the DeviceType is "Database" or "Operating System"
- Do not allow if SafeName or PolicyID has PCI in their name

If the property name is prefixed with **REGEX:** the value will be used as a regular expression comparison, otherwise it will be used as an exact match: