DAP最新机能配置示例

目录

简介

本文描述动态访问策略(DAP)的最新机能远程访问VPN的。当您需要另外的灵活性由标准时，匹配您能使用这些最新机能。

> **注意**：使用 **debug** 命令之前，请参阅有关 Debug 命令的重要信息。

先决条件

要求

Cisco 建议您了解以下主题：

- 对基本DAP的了解要求。参考ASA 8.x动态访问策略(DAP)部署指南(支持文档)或ASA 8.x动态访问策略(DAP)部署指南(支持社区)。
- 一好了解Lua编程也是有利的。参考Lua编程的材料可用在Web。

使用的组件

本文没有限制对特定软件和硬件版本，但是可适应安全设备管理器(ASDM)要求为了完成配置。

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您使用的是真实网络，请确保您已经了解所有命令的潜在影响。

背景信息

> **警告：**请使用DAP提前的自定义Lua功能，只有当ASDM GUI配置或EVAL功能不提供您需要的匹配的行为。在生产部署，请使用先进的Lua功能与非常小心和以从Cisco工程师/技术支持中心(TAC)的指导为了避免与DAP的所有不愿意的行为。

如果使用DAP远程访问VPN，您可能需要另外的灵活性由标准匹配。例如，您能应用根据这些方案的不同的DAP ：

- 组织单位(OU)或层级的其他级别包含用户对象。
- 组名(memberOf)遵守命名规则，但是有许多可能的匹配，因此您要使用在组名的一个通配符。
- 您要检查抗病毒，反间谍软件或者防火墙包在终端PC。

1. 请使用ASDM为了创建您的匹配标准的一个逻辑表达式。

2. 请使用Advanced模式为了创建与一个逻辑表达式和Lua代码的自定义函数。

基于OU的匹配示例

轻量级目录访问协议(LDAP)服务器能返回DAP在一个逻辑表达式能使用的许多属性。

对于这些属性示例，请使用**调试dap trace命令**在可适应安全工具(ASA)控制台。

```
assert(function()
if ( (type(aaa.ldap.distinguishedName) == "string") and
(string.find(aaa.ldap.distinguishedName, "OU=Admins,dc=cisco,dc=com$")
```

```
~= nil) ) then
return true
end
return false
end
```

用户的特有名(DN)是LDAP服务器返回的一个属性。DN隐含地识别用户对象在目录的地方查找。例如，如果DN是CN=Joe用户， OU=Admins， dc=cisco， dc=com，此用户在OU=Admins查找， dc=cisco， dc=com。如果所有管理员是在此OU (或任何容器在此级别之下)，请使用此逻辑表达式为了匹配在标准：

```
assert(function()
if ( (type(aaa.ldap.distinguishedName) == "string") and
(string.find(aaa.ldap.distinguishedName, "OU=Admins,dc=cisco,dc=com$")
~= nil) ) then
return true
end
return false
end)()
```

在本例中， string.find功能允许常规表示。$在字符串结束时停住此字符串对distinguishedName字段的末端。

请使用转义字符%在您的搜索字符串为了退出特殊字符例如()。% + - * ？[^ $。例如，您能退出在此字符串(OU=Admins的-字符， dc=my-domain， dc=com$)如此字符串(OU=Admins所显示， dc=my%-domain， dc=com$)。

组成员(memberOf)示例

您能创建激活目录(AD)组成员模式匹配的一个相似，基本逻辑表达式。由于用户可以是多个组的成员， DAP在表里解析从LDAP服务器的答复到独立的条目并且拿着他们。在这种情况下，最新机能要求为了：

- 如果用户只属于一组，比较memberOf字段作为字符串。
- 重复执行通过每个返回的memberOf字段，如果返回的数据是类型‘表’。

例如，如果用户是结束以‘任何组的成员- stu’，他们匹配此DAP ：

```
assert(function()
local pattern = "-stu$"
local attribute = aaa.ldap.memberOf
if ((type(attribute) == "string") and
(string.find(attribute, pattern) ~= nil)) then
return true
elseif (type(attribute) == "table") then
local k, v
for k, v in pairs(attribute) do
if (string.find(v, pattern) ~= nil) then
return true
end
end
end
return false
end)()
```

与自定义函数的CheckAndMsg

此功能以设置的操作使用DAP终止：

```
(assert(function()
local block_connection = true
local update_threshold = "150000" --this is the value of lastupdate in
seconds
for k,v in pairs(endpoint.av) do
if (CheckAndMsg(EVAL(v.exists, "EQ", "true", "string") and EVAL
(v.lastupdate, "LT", update_threshhold, "integer"),k.." exists; last update is
"..string.sub((tonumber(v.lastupdate)/86400), 1, 3).." days",k.." does not exist; last update is
"..string.sub((tonumber(v.lastupdate)/86400), 1, 3).." days")) then
block_connection = false
end
end
return block_connection
end)())
```

在终端，它显示此消息：

```
Login denied.<AV Name> does not exists; last update is <X> days
```

防病毒、反间谍软件和防火墙示例

这些Lua功能检验属性涉及到抗病毒，反间谍软件和防火墙包在Cisco Secure Desktop (CSD)主机扫描返回的终端PC。

检查抗病毒安装

此自定义函数检查CSD是否检测中的任一抗病毒：

```
assert(function()
for k,v in pairs(endpoint.av) do
if (EVAL(v.exists, "EQ", "true", "string")) then
return true
end
end
return false
end)()
```

检查抗病毒安装和最近一次更新，并且提供错误消息

此示例展示DAP如何能检查抗病毒安装，检查最近一次更新和为修正通知用户。它使用一个功能类似于那在控制中抗病毒安装。

设置验证、授权和统计(AAA)归因于您希望匹配。在Advanced字段，请保证这和操作选择;在战场，请保证终止选项选择。如果用户匹配AAA属性，并且，如果Lua功能返回值真， DAP选择，解释的消息出现DAP记录为什么显示，并且用户连接终止。如果Lua功能不返回值真， DAP不配比和许可证访问。在消息框领域，请输入消息， “没有抗病毒程序找到，再请安装抗病毒和尝试”。如果用户有一个抗病毒包并且是在更新天阈值之下，他们没有给消息如表示由在线路此示例7的双引号：

```
(assert(function()
local block_connection = true
local update_days = "15" --days
local av_lastupdate = update_days*86400
for k,v in pairs(endpoint.av) do
if (CheckAndMsg(EVAL(v.exists, "EQ", "true", "string") and EVAL(v.lastupdate, "LT",
```

```
av_lastupdate, "integer"),"",k.." exists; but last update is greater than 15 days old. Expecting
under 15 days.")) then
block_connection = false
elseif (EVAL(v.exists, "NE", "true", "string")) then
block_connection = true
end
end
return block_connection
end)())
```

如果用户有抗病毒的Norton，但是最近一次更新比15天极大，此示例消息出现：

```
NortonAV exists; but last update is greater than 15 days old. Expecting under 15 days.
```

如果EVAL不配比，去下个功能，匹配，并且返回值真。因为没有CheckAndMsg关联与第二个功能，使用DAP文本：

```
No anti-virus program found, please install anti-virus and try again.
```

总之， DAP寻找一个用户AAA和终端属性为了匹配DAP。如果DAP配比，用户终止与消息。终端匹配是返回真或错误对DAP Lua EVAL的结果。真的匹配和拒绝连接。错误不匹配和允许连接。

1. 在循环检测的第一个功能endpoint.av.xxxxx.exists是否与真是相等的，并且最近一次更新是否比已配置的天是较少。没有防病毒软件的用户是允许的访问，因为用户AAA配比，但是Lua特别地查找为endpoint.av.xxxxx.exists =真和endpoint.av.xxxxx.lastupdate <=天。
2. 因为第二个功能为真， endpoint.av.xxxxx.exists的NE仅查找第二条环路捉住用户，不用防病毒软件并且阻塞他们。如果终端av存在的用户与真不是相等的，功能返回值真，含义他们没有抗病毒。DAP匹配并且拒绝连接。

检查反间谍软件安装

此自定义函数检查CSD是否检测反间谍软件：

```
assert(function()
for k,v in pairs(endpoint.as) do
if (EVAL(v.exists, "EQ", "true", "string")) then
return true
end
end
return false
end)()
```

检查防火墙安装

此自定义函数检查CSD是否检测防火墙：

```
assert(function()
for k,v in pairs(endpoint.fw) do
if (EVAL(v.exists, "EQ", "true", "string")) then
return true
end
end
return false
end)()
```

检查防病毒、反间谍软件或者防火墙安装

如果找到，此功能返回真抗病毒，反间谍软件或者防火墙包：

```
assert(function()
function check(antix)
if (type(antix) == "table") then
for k,v in pairs(antix) do
if (EVAL(v.exists, "EQ", "true", "string")) then
return true
end
end
end
return false
end
return (check(endpoint.av) or check(endpoint.fw) or check(endpoint.as))
end)()
```

终止，如果没有间谍软件安装

在此功能和功能之间的唯一的区别在控制中反间谍软件安装是‘没有’先于主张。

```
not assert(function()
for k,v in pairs(endpoint.as) do
if (EVAL(v.exists, "EQ", "true", "string")) then
return true
end
end
return false
end)()
```

比三十天检查防病毒和防火墙安装，并且验证最后抗病毒更新不极大

此示例返回真，如果找到抗病毒和防火墙，并且，如果最近一次更新抗病毒比30天不极大：

```
assert(function()
function checkav(antix)
if (type(antix) == "table") then
for k,v in pairs(antix) do
if (EVAL(v.activescan, "EQ", "ok", "string") and EVAL (v.lastupdate, "LT", "2592000",
"integer")) then
return true
end
end
end
return false
end
function checkfw(antix)
if (type(antix) == "table") then
for k,v in pairs(antix) do
if (EVAL(v.enabled, "EQ", "ok", "string")) then
return true
end
end
end
return false
end
return (checkav(endpoint.av) and checkfw(endpoint.fw))
end)()
```

由于防火墙没有返回一个lastupdate的值，有一个分开的功能。

常规表示匹配

此部分描述使用REGEX表达式为了匹配某些属性和确定主机的正确性的功能。这些REGEX功能测试了并且有效：

- 美元的符号($)停住搜索字符串对返回值的结尾。
- 脱字号(^)停住搜索字符串对返回值的开始处。
- 托的字符，例如[Aa]，匹配多个字符在一个特定位置。例如，为了匹配Ou=cisco (不区分的案件)，使用OU= [Cc] [Ii] [Ss] [Cc] [Oo]。
- 期限(。)在此位置匹配所有单个字符。例如，组。用户匹配Group01Users，Group33Users，等等。

如果终端PC有ICM Hotfixes KB944，任何实例请连接

此功能使用匹配的常规表示为了发现ICM Hotfixes列表是否包含模式。在本例中， Cisco Secure Desktop返回在终端PC的所有ICM Hotfixes;如果有KB944实例， DAP策略配比和被强制执行。

```
assert(function ()
local pattern = "KB944"
local true_on_match = true
local match = false
for k,v in pairs(endpoint.os.hotfix) do
print(k)
match = string.find(k, pattern)
if (match) then
if (true_on_match) then
return true
else return (false)
end
end
end
end)()
```

例如，如果主机有ICM Hotfixes KB944533或ICM Hotfixes KB944653，它匹配规则。

检查MAC地址OUI的使用LUA脚本

如果终端PC有ICM Hotfixes KB944，任何实例此功能类似于在连接描述的那个。此功能使用常规表示为了组织匹配唯一标识符(OUI) MAC地址。

在本例中， MAC地址从d067.e5XX.XX开始。请使用一个常规表示和Lua代码为了匹配从同样OUI MAC启动的机器。

```
assert(function ()
local pattern = "^d067\.e5*"
local true_on_match = true

local match = false
for k,v in pairs(endpoint.device.MAC) do
print(k)
match = string.find(k, pattern)
if (match) then
if (true_on_match) then
return true
else return (false)
end
end
end
end)()
```

注意：此功能一个不同的版本为一多值检查要求。

根据主机名的前三个字母的连接(不区分大小写)

此功能使用常规表达为了确定主机名的前三个字母是否是msv (不区分的案件) ：

```
assert(function()
local match_pattern = "^[Mm][Ss][Vv]"
local match_value = endpoint.device.hostname
if (type(match_value) == "string") then
if (string.find(match_value, match_pattern) ~= nil) then
return true
end
elseif (type(match_value) == "table") then
local k,v
for k,v in pairs(match_value) do
if (string.find(v, match_pattern) ~= nil) then
return true
end
end
end
return false
end)()
```

如果终端PC和序列号Device.id在证书是相同的，请连接

如果终端PC和序列号的device.id在证书是相同的，此Lua表达式被认为连接：

```
assert(function()
local match_pattern = endpoint.device.id
for k,v in pairs(endpoint.certificate.user) do
if (type(v.subject_e) == "string") then
if (string.find(v.subject_e, match_pattern) ~= nil) then
return true
end
elseif (type(v.subject_e) == "table") then
local k,v
for k,v in pairs(v.subject_e) do
if (string.find(v, match_pattern) ~= nil) then
return true
end
end
end
return false
end ) ()
```

注意：使用通配符(*)在此特定功能不工作(endpoint.certificate.user ["*"]不工作)。您必须单个获取每个KV对和通过他们解析。

强制执行根据CSD域注册表项的主机扫描的DAP

此步骤提供配置程序的示例ASDM。

1. 找出保持域在\ HKEY_LOCAL_MACHINE \系统\ Currentcontrolset \服务\ Tcpip \参数\域的注

册表项。

2. 定义注册表设置的主机扫描参数。

3. 适用注册终端属性对DAP策略。

4. 建立安全套接字协议层(SSL) VPN会话。

5. 通过DAP调试验证DAP策略执行。

DAP为Windows 7和CSD 3.5支持

Windows 7平台支持与CSD版本3.5或以上。使用ASDM 6.2.x维护版和6.3.x版本，您能直接地使用接口为了检查Windows 7 OS。使用初期的ASDM版本，一份先进的DAP Lua脚本要求为了检查Windows 7台机器。在ASA用版本8.x和PRE BETA CSD版本3.5，请输入此Lua脚本字符串到ASDM DAP提前的方框为了实行检查Windows 7台机器：

```
(EVAL(endpoint.os.version,"EQ","Windows 7","string"))
```

IP电话、iPads和移动设备的识别

此Lua表达式让您由他们的唯一标识符(UIDs)跟踪特定移动设备。您能使用DAP为了达到此基本功能。

当值不可以是硬编码和需要读从AD时，这变得更加困难。由于没有AD的特定UID字段，您能存储特定用户的值在不同字段下。此示例使用otherHomePhone存储UID。

要帮助您识别IP电话或iPad的UID，请搜索Web一个适当的工具。

一旦识别UID，请添加它到在AD条目的otherHomePhone该用户的。

从**调试ldap 255**命令和从用户测试验证，请得到LDAP属性推送的，是otherHomePhone。

允许电话连接，然后在已尝试连接时运行DAP trace为了识别包含UID的终端属性(endpoint.anyconnect.deviceuniqueid)。

此Lua表达式能然后比较两个参数：

```
assert(function()
if (type(aaa.ldap.otherHomePhone) ==type(endpoint.anyconnect.deviceuniqueid)
then
return true
end
```

```
return false
end)()
```

使用DAP防止连接由特定浏览器

此步骤描述如何使用DAP由镀铬物浏览器防止连接：

1. Enable (event) CSD。

2. 在主机扫描配置下，请使用进程ID (PID)和进程名为了添加进程扫描。

   您能确定PID和进程名在Windows与任务管理器。为了显示PID值，打开**任务管理器**，去**Processes选项**，点击**视图菜单**，然后单击**选择列**。在挑选列或请选择进程页列对话，做标记并且检查复选框**PID (进程标识符)**，并且点击OK键。

   在橡皮防水布上，您能确定进程ID用活动监控程序。或者，在您在Unix能能也使用)的bash shell (请使用**ps - e**命令，当进程运行时，然后匹配PID到进程名用**cat /proc/ <PID>/cmdline**命令。

3. 如果该进程在计算机，运行创建DAP策略为了测试，尤其。

4. 测试您的连接。

警告

1. 与此解决方案的问题是用户不能有镀铬物开放在他的计算机。DAP检查该特定的镀铬物进程是否运行，但是不检查发现无客户端会话是否启动由该进程或由某其他进程。因此，当WebVPN连接尝试时，用户不能运行在背景的镀铬物。
2. 假设用户使用Firefox开始WebVPN会话的一个方案和登录尝试出故障。用户记得镀铬物仍然运行，因此用户终止镀铬物连接并且设法再登录。因为CSD必须重新运行主机扫描，登录仍然发生故障。因此，用户必须也结束使用访问WebVPN Firefox的实例，然后重新启动Firefox。此进程可以是混乱的对用户。思科建议您创建告诉用户终止镀铬物和关闭浏览器他们当前使用的DAP故障消息：

FAQ

此部分提供一答案到多数常见问题之一看待在本文描述的信息。

LUA为什么写脚本工作为一些用户，但是不为所有？

考虑此LUA脚本：

```
    assert(function()
         for k,v in pairs(endpoint.certificate.user) do
             if (v.subject_store == "capi" and v.subject_dc == "homedepot") then
                 return true
             end
         end
         return false
     end)()
```

此脚本设计为了匹配证书存储和在证书被找到的主题DC。然而，此脚本用多台机器在一些机器在许多其他测试和被发现工作，但是失效。

原因此脚本只运作间歇地是由于hostscan返回值的方法。当不工作的您查看DAP trace时，您能看到*subject_dc*返回多个值每证书。您能也看到返回的不是*家得宝*：

```
DAP_TRACE: endpoint.policy.location = "CORP-Windows"
:
.
DAP_TRACE: endpoint.certificate.user["6"].subject_store = "capi"
DAP_TRACE: endpoint.certificate.user["6"].subject_dc = "com"
DAP_TRACE: endpoint.certificate.user["6"].subject_dc = "homedepot"
DAP_TRACE: endpoint.certificate.user["6"].subject_dc = "amer"
```

当工作的您查看DAP trace时，这可以被观察：

```
DAP_TRACE: endpoint.certificate.user["20"] = {}
DAP_TRACE: endpoint.certificate.user["20"].subject_cn = "JHD0C6"
DAP_TRACE: endpoint.certificate.user["20"].subject_e = "jimmie_harden@homedepot.com"
DAP_TRACE: endpoint.certificate.user["20"].subject_ou = "Associates"
DAP_TRACE: endpoint.certificate.user["20"].subject_store = "capi"
DAP_TRACE: endpoint.certificate.user["20"].subject_dc = "com"
DAP_TRACE: endpoint.certificate.user["20"].subject_dc = "homedepot"
DAP_TRACE: endpoint.certificate.user["20"].issuer_cn = "The Home Depot Remote
Access Issuing CA v2"
```

这表明LUA脚本适当地运作。然而，由于状态评估返回在一些机器的值的方法，脚本不配比。在这种情况下， Cisco Bug ID的CSCuu85646修正和CSCuh67472要求。