APM: Break it down Yo!

Josh Michaels, 2012-23-08

Access systems are messy. Wait, let me rephrase that, Poorly planned access systems are messy. We've all seen it happen a thousand times. Someone comes running into the security cave

Demanding Admin: "We have a new web property going up and I need to give remote admin access to the team at the north pole! Santa must have access!"

Security Monkey: "Ok, does anyone else need access?"

Demanding Admin: "Nope, just Santa.. oh and his elves. And maybe the Polar bear consortium. They are going to be blogging about the entire web experience"

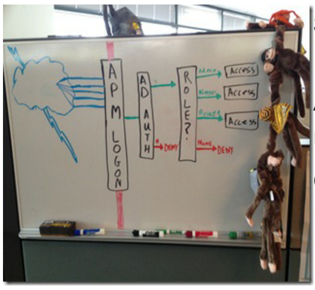
Security Monkey: "Anyone else?"

Demanding Admin: "Nope, not at all. For sure. Well, except for the secret project group A, but they only should have access to the subterminal C. And we need to be able to create new users and eliminate old ones from the main auth system"

Security Monkey:

"Apprehensive monkey is apprehensive"

It's at this point, that we have should pick up the marker and start walking the system. It always makes more sense when drawn out. Well, most of the time. Sometimes it ends up like a Picasso-like portrait that belongs in the Louvre.



So here is what we end up with.

All users hit the APM login page.
User logs in at **AD AUTH**:
Good credentials –> proceed to ROLES
Bad Credentials –> DENY

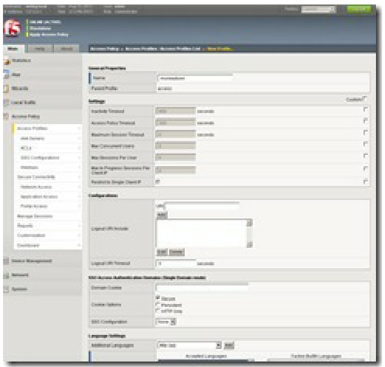
ROLES (check AD to find the user's groups):
Admin –> Allow full admin access/resources
Blogger –> Give a protected path to blogs
Project A –> Allow into secret project A
No role –> DENY!

Easy Peasy Lemon Squeezy! Lets get to it!

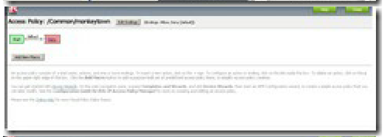
Making it Happen Captain:

Access the APM, click on Access Profiles, can click Create.

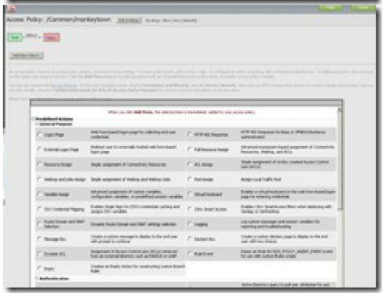
For our policy, we are going to give it a name and add english as the accepted language.



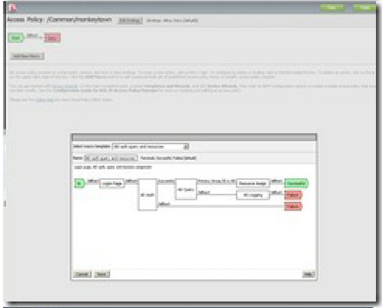
Now, Click edit on the policy and let the fun begin.



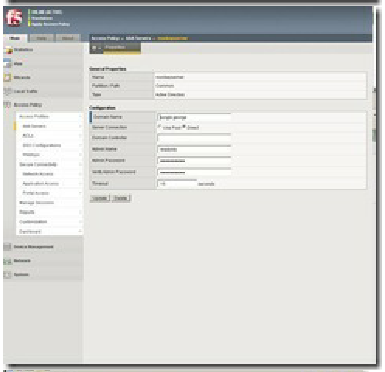
We click the plus sign and start picking the access path items. Logon Page first.



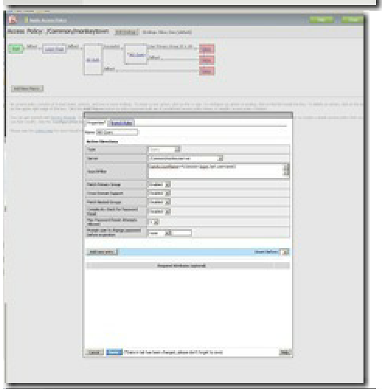
Next, we know we want to Authenticate the user against AD. We will need to specify an AD server here.



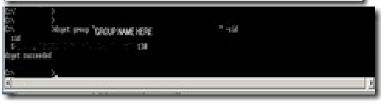
Define the AD Server that will be used here:
Access Policy –> AAA Servers –> Active Directory. Define a Name, the domain that users belong to, and give it a domain guest account to connect with.



Now, we go back and add the ROLES Query. In the APM terms, we want an AD Query.   We will need to tell it a search filter. For us, we'll use the logon name: (samAccountName=%{session.logon.last.username}) And, since we plan on setting the user's primary groups, we'll tell the query to fetch those.



Now to set the success paths, depending on group. Here we click on branch rules under the AD Query. We need to know the group ID for this. You can ping your AD admins, or if you have access, run the query against the AD Server:

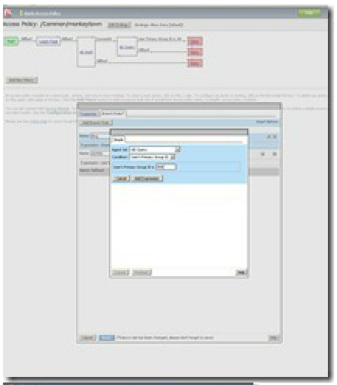


```
dsget group "GROUP NAME HERE                    " -sid
  sid
  S                                   18
dsget succeeded
```

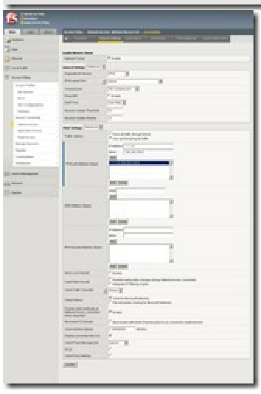
```
C:\me>dsget group "CN=Group here" -sid
  sid
  S-1-1-11-000000000-0003009999-9876543211-130
dsget succeeded
```

The groups ID is: 130

We build all 3 branches here based on the Primary groups. NOTE: we can also do “member of” or any number of other conditions.

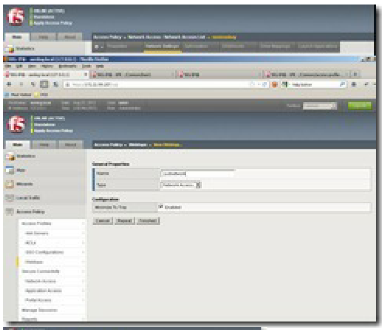


Now we need to think about who gets what resources. Based on the requirements we are going to build the following:

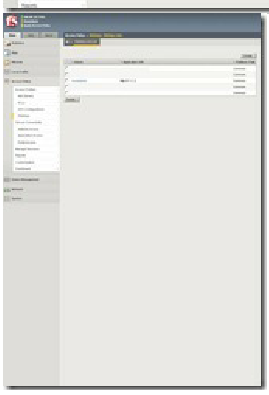


Admin: Full webtop/Network Access
Blog: Simple Connectivity Access
Secret Project: Directed Network access

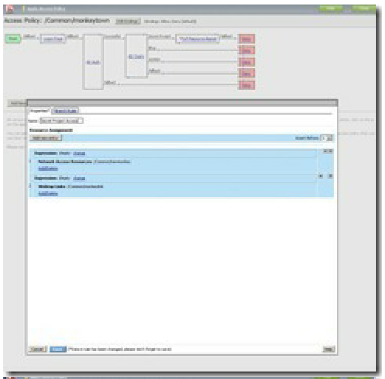
Start with the Network Access Settings:
Access Policy –> Network Access –>Create
Once you’ve named it, you have to edit the policy to define the network path. For us, we want a split tunnel (only send traffic for 1.1.1.0/24 through the tunnel).
Now we need a Network webtop for our Blog/Project Access. Click on Webtops->create. Add a name and select “Network Access”



We also need a Full webtop for our admins. This creates an actual interactive webtop of all resources they can use. It’s a container for any Webtop Links we create (under Webtops->Links)

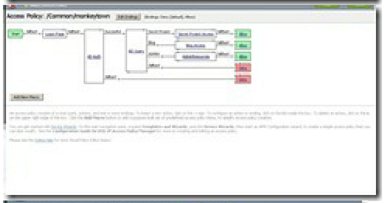


Sweet, now we can go back to the policy editor and assign out rights. At this point, we just add a Full Resource assignment to each path, and select the resources we want them to access.
Admin: Full webtop with links
Blog: monkeylink and monkeylink network access
Secret: same as blogger, for now.



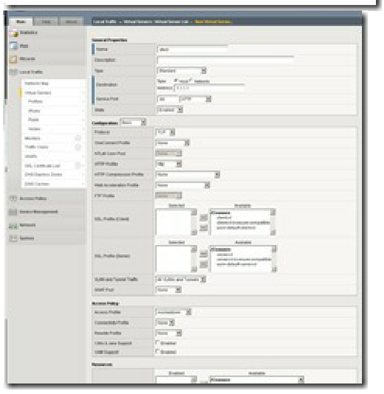
Access Policy: /Common/monkeytown

Last Policy change is to set the fallback for the success paths to Allow.



Access Policy: /Common/monkeytown

Finally, we apply the access policy to a virtual server, and boom, APM in place.



So, it takes a bit of clicking to put it all in place. But once you have the foundation laid down, you can easily add new paths or make changes to current access roles. I highly recommend that anytime you are going to make a large change, copy the current policy, and edit on the copy (better safe than sorry).

F5 Networks, Inc. | 401 Elliot Avenue West, Seattle, WA 98119 | 888-882-4447 | f5.com

F5 Networks, Inc.
Corporate Headquarters
info@f5.com

F5 Networks
Asia-Pacific
apacinfo@f5.com

F5 Networks Ltd.
Europe/Middle-East/Africa
emeainfo@f5.com

F5 Networks
Japan K.K.
f5j-info@f5.com