Microsoft NPS - Improving reliability of as an authentication provider for eduroam

August 1st 2014

An article written by Simon Edwins of measeures he investigated to improve the performance and reliability of the organisation's RADIUS service to support eduroam - including adjusting the Max Concurrent Api setting:

> I've spent a fair bit of time over the past month trying to improve the reliability of our RADIUS service for eduroam. Previously it was entirely based on Microsoft NPS which has the tendency to silently discard authentication packets which it should really be rejecting. This creates a problem because if the authentication request originated from outside of your network (i.e. a roaming user authenticating from a remote organisation) the RADIUS server will appear to be non-responsive as far as the JANET NRPS is concerned and will be marked as offline for 300s. Once your RADIUS servers are marked as offline legitimate requests will not be sent to them and will fail. As much as I would have liked to replace the NPS servers with FreeRADIUS altogether, they do integrate with some other services we provide written for NPS so that wasn't possible.
>
> I spent a fair bit of time trying to work out why our NPS servers were discarding packets at all. In case it's of any use to anyone, I made a few discoveries and subsequent changes to our RADIUS infrastructure to try and alleviate the problems, as described below:
>
> 1. NPS can discard RADIUS authentication requests if they contain invalid attributes. It seems to depend upon how NPS determines whether the request is invalid as to whether it rejects or silently discards the request. Since there is no attribute filtering ability within NPS I ultimately decided to create two new FreeRADIUS servers which function purely as proxy servers between our NPS servers and the JANET NRPS. I filtered out all but the following inbound (remembering that NPS doesn't recognise operator-name):
>
> ```
> User-Name =* ANY,
> Reply-Message =* ANY,
> State =* ANY,
> Class =* ANY,
> Message-Authenticator =* ANY,
> Proxy-State =* ANY,
> EAP-Message =* ANY,
> MS-MPPE-Send-Key =* ANY,
> MS-MPPE-Recv-Key =* ANY,
> Calling-Station-ID =* ANY,
> ```

Chargeable-User-Identity =* ANY,
NAS-IP-Address =* ANY,
Framed-MTU >= 576,
NAS-Identifier =* ANY

We're being a good eduroam institution and we also filter outbound RADIUS messages to only contain the following attributes:

User-Name =* ANY,
Reply-Message =* ANY,
State =* ANY,
Class =* ANY,
Message-Authenticator =* ANY,
Proxy-State =* ANY,
EAP-Message =* ANY,
MS-MPPE-Send-Key =* ANY,
MS-MPPE-Recv-Key =* ANY,
Calling-Station-ID =* ANY,
Operator-Name =* ANY,
Chargeable-User-Identity =* ANY,
NAS-IP-Address =* ANY,
Framed-MTU >= 576,
NAS-Identifier =* ANY

This also prevents stray VLAN attributes from reaching our NPS servers which was causing us some problems if we were visited by users from other organisations that sent access-accept messages with VLAN information which is only valid at the user's home site.

The FreeRADIUS proxy servers also allow us to inject the Operator-Name attribute into outbound RADIUS packets. This was not possible with NPS either.

2. Unfortunately we were still seeing many silent discards on our NPS servers which were causing timeouts on the JANET NRPS. As a temporary solution I resorted to configuring FreeRADIUS to send back access-rejects if our NPS servers did not respond within 8s (since the NRPS timeout seems to be 10s). In tandem with this I configured FreeRADIUS to send authentication requests to test the availability of the NPS servers as soon as they were found to be non-responsive (zombies in FreeRADIUS speak). This ensured I didn't break failover of our NPS servers.

3. Next I focused on trying to fix the discard errors on NPS. This was not so easy. Firstly, we were seeing events discarded with the error message "Authentication failed due to an EAP session timeout; the EAP session with the access client was incomplete". These were mainly caused if the wireless device was connected with a very poor signal strength which prevented the full EAP communication from taking place. The solution to this problem was to disable the lowest data rates available to clients connecting to eduroam on our access points. I found that disabling the 1.0 and 2.0 data rates was sufficient to clear up 99% of the failures from internal devices. Most of the events that remain relate to connections at other organisations so there's nothing I can do about those.

4. Most of the other discards had the error message "An internal error occurred. Check the system event log for additional information." (don't you just hate those, especially when the system event log had nothing logged!). It turns out that almost all of these were caused by some Apple and Android devices connecting with bad passwords. We had the number of re-authentication attempts set to 3, and with this configuration in place a device would offer the user the option of changing their password. If they entered the correct password it would still fail to connect because the NPS servers would discard the events. It looks like the requests were discarded because the "SequenceID" in the RADIUS packet hadn't been incremented. Not all devices behave that way and the only workaround I've found so far is to change the re-authentication attempts to 0. It's not an ideal solution, but at least it stops the NPS server discarding these events as it starts rejecting instead.

I believe we do still get a small number of discards each day, and these I'll look into as they arise.

5. To try and improve the performance and availability of our NPS servers I've made a couple of other registry changes, as follows:

Set HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL to 3. This increases SCHANNEL logging.

Set HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Netlogon\Parameters\MaxConcurr to 5. This increases the available authentication threads to the DCs since our RADIUS servers are not Domain Controllers.

According to the JANET NRPS logs it seems that the only remaining errors we have relate to users with typos in their username or with bad or expires passwords. To try and reduce the problem we're producing daily RADIUS failure reports which are now being reviewed by our helpdesk staff so they can contact the worst offenders. Of course, it's impossible to remove all these failures but anything is better than nothing.

See also this article from Microsoft that explains what tuning you can do with MaxConcurrentApi - this might help with possible authentication bottlenecks:

https://support.microsoft.com/en-gb/help/2688798/how-to-do-performance-tuning-for-ntlm-authentication-by-using-the-maxc [1]

Links
[1] https://support.microsoft.com/en-gb/help/2688798/how-to-do-performance-tuning-for-ntlm-authentication-by-using-the-maxc