

**Support SAML SSO for VIA Users**

**Revision 0.6**

Copyright © 2016 Aruba Networks, Inc. All rights reserved. All computer programs referred to in this document contain the confidential and intellectual property of Aruba Networks, Inc. No part of this form may be used, copied, disclosed, or transferred to any party without prior written authorization from Aruba Networks, Inc. This document is uncontrolled when printed.

Definitions

AOS Aruba OS

Auth Authentication Manager Process

CPPM ClearPass Policy Manager

RADIUS Remote Dial-in User Server that provides AAA management

Role User-role, containing a set of policies to describe the client’s network privileges

VSA Vendor Specific Attributes

SSO Single sign on

SAML Security Assertion Markup Language is a standard for exchanging authentication and authorization data between parties

SP Service Provider does access control decision based on the assertion response received from IDP

IDP Identity Provider authenticates the client and provides the assertion response

Token/tunnel-cookie Is compressed, encoded, possibly encrypted opaque data used to identify the session/user

TABLE OF CONTENTS

Definitions 2

TABLE OF CONTENTS 3

1 Introduction 4

2 Rationale/Deployment use case 4

3 Functional Description 4

4 Proposed Implementation 4

4.1 VIA fetch Authentication Profile names 5

4.2 VIA SAML SSO login 5

4.2.1 VIA downloads the tunnel\_cookie 5

4.2.2 VIA login using tunnel\_cookie 8

4.2.3 Example configuration on controller: 8

4.2.4 Need for tunnel-cookie token 9

4.3 VIA PROFILE DOWNLOAD 9

4.4 VIA IKE/IPSEC CONNECTION 10

4.4.1 Need for External ACL Server 12

4.4.2 Radius Request attributes during XAUTH 12

4.4.3 Query for downloadable role definition 12

4.4.4 External ACL server Response formatC 12

4.5 ROLE DERIVATION 13

4.5.1 Sample McDonald Role specification 13

5 Controller-side Configuration 15

5.1 Enable SSO for VIA 15

5.2 CPPM SSO redirect URL 15

5.3 External ACL Server configuration 15

6 Administration 15

6.1 CLI 15

6.2 UI 16

7 Licensing 16

8 Migration Strategy 16

9 Interoperability 16

10 Monitoring 16

11 Performance and scale 16

12 Unit tests and Guidelines for QA 17

13 Open Issues 17

14 Deferred Features 18

15 Dependencies 18

16 Meeting Minutes 18

17 Appendix 19

# Introduction

This document describes the proposed implementation to support SAML SSO for VIA users and to control N/W access based on the attributes in the SAML response issued by IDP.

N/W access is controlled by the ACLs/ACEs on the Controller, which are derived from the SAML assertion attributes defined in the SAML response.

This design should work with external third-party SAML IDP and SAML assertion defined by McDonald’s.

# Rationale/Deployment use case

The functional requirement from McDonald’s is to provide secure remote inbound access for its vendors into the store network. These vendors would like to have remote access to other infrastructure components installed within the store (like cameras). This secure inbound access is being implemented through the VIA client terminating a VPN session into 7240s at the AT&T DC (acting as the VIA Stack). For authenticating the users, McDonald’s would like to use their own third party IDP.

# Functional Description

As a part of this enhancement, we need to meet the below requirements:

1. When the user submits username/password through the VIA client, the VIA client should be redirected to the external IDP to be authenticated against.
2. CPMM could be used as SP to off-load SAML from controller
3. The VIA client initiates the IKE session to the controller and provides SAML assertion or token from CPMM to authenticate the user without prompting for credentials again
4. Based on the attributes returned during the SAML assertion, the controller should get the role/dynamic ACLs for the user
5. Once the VIA user disconnects downloadable-roles and corresponding ACLs/ACEs need to be removed.

This functionality requires SAML based authentication and downloadable-roles to be applied to VIA client users. Today SAML-based authentication is available for dot1x authenticated users. This feature is to extend SAML-based authentication to VIA users and apply downloadable-roles based on the attributes returned during SAML assertion.

# Proposed Implementation

SAML-based solution has two major building blocks SP and IDP. SP (Controller) facilities the services to the client when it presents a valid SAML response issued by the IDP. Couple of approaches been proposed to provide the functionality: Firstly, Controller needs to understand SAML specification to validate the SAML response and will be the SP. Secondly, Controller could off-load the SAML-awareness to another n/w entity like CPPM and consult CPPM for SAML assertion validation functionality. The rest of the document is regarding design of the second approach. Refer SECTION- 19 for details about first approach.

For reasons below, the design to off-load the SAML-awareness to CPPM is considered by all teams and will be implemented

1. By off-loading SAML-awareness to CPPM, controller changes could be minimized. Also, this enables to build SSO solution with controller being agnostic of underlying technology like SAML/OAUTH.
2. Today CPPM already supports SAML and it is integral part of existing downloadable role feature that is supported for VIA users.

## VIA fetch Authentication Profile names

1

|  |
| --- |
| VIA client  Controller  (1)VIA login https request to <controller>/via  (2)200 OK, with response containing the list of VIA Authentication profiles. Profiles could include authentication with user credentials/certificate/SSO. Each profile in the list will be qualified with the authentication type. |

## VIA SAML SSO login

To support SSO way of login for VIA clients, VIA clients will have to get the SAML tunnel\_cookie and use this to login to the controller. This is done in two steps.

* Firstly, VIA client will initiate a request to <controller>/via/sso (without the sso\_user/sso\_cookie) and this will result in tunnel\_cookie to be downloaded as described below in Section 4.2.1.
* Secondly, once the tunnel\_cookie is available VIA client will initiate a login attempt to <controller>/via/login along with the above sso\_user/sso\_cookie as described in Section 4.2.2.

### VIA downloads the tunnel\_cookie

VIA client sends a HTTPS POST request to <controller>/via/sso:port, with vap:<via-auth-profile-name-sso> when SSO is enabled. Port used here will be existing default port used for VIA client authentication with user credentials. Controller will respond with the 302 status and redirect URL to CPPM (URL is configured on controller). By end of this step, VIA client will have tunnel-cookie and using this VIA client will attempt to login first.

Tunnel-cookie that CPPM generates and provides to the VIA client should be saved by CPPM for 300 secs. During this time period VIA client should be able to login or download the profile or establish IKE/IPSEC tunnel without being prompted for credentials again.

Please note Controller is still the SP for providing the N/W access services and CPPM is acting as an SP for SAML protocol related functionalities only.

#### Need for STEP-11 in Flow-diagram

VIA will be using a WPF (Windows Presentation Foundation) browser control to allow user to enter username/password in SAML IDP login page. Because of this VIA client can’t directly get the sso\_user/sso\_cookie fields that gets POSTed to controller URL. In

STEP-12 of below flow-diagram, the javascript code contained in the CPPM final response page will do a POST to URI “/via/sso/parse” in controller with sso\_user and sso\_cookie.

The controller implementation of the handler for the URI “/via/sso/parse” needs to extract these parameters from the POST and return them in the response for VIA client to pick up.

|  |
| --- |
| (5) Initiate a SAML request with redirect to IDP  (2)VIA login https request to <controller>/via/sso  with vap=<via-auth-profile-name-sso>  (12)Javascript in CPPM response page POSTs to controller at “/via/sso/parse” with params “sso\_user” “sso\_cookie”  (4) HTTPS session to CPPM URL  (13)Controller responds with cookie “sso\_user” and “sso\_cookie”  (11) CPPM parses the above SAML response and caches the store-id in it and generates the cookie. CPPM sends back Response page with Javascript.  (3)Sends back response with 302 Redirect to CPPM URL.  (10) HTTPS with SAML response to the CPPM/SP  (9)IDP authenticates and sends back SAML response, 302-Redirect to SP with status code and other attributes  (6) HTTPS with SAML request to IDP  (14) VIA extracts “sso\_user” and “sso\_cookie”  Controller  VIA client  CPPM  IDP  (7)IDP throws back page asking for credentials  (8)User enters username/password and submits  WPA Browser Widget  (1)WPA browser Widget |

### VIA login using tunnel\_cookie

Once the VIA client gets the tunnel\_cookie as in Section 4.2.1, It will initiate a login request to the controller /via/login with Auth profile name and sso\_user/sso\_cookie as vap and uid/passwd (this is as existing feature). Controller sends a radius authentication request to CPPM with sso\_user and sso\_cookie and CPPM will reply with role configured. Depending on this role being present on the controller at that point, appropriate role is assigned to the VIA user as explained in Section 4.2.3.

|  |
| --- |
| VIA client  Controller  (1)VIA login https request to <controller>/via/login with vap= <via-auth-profile-name-sso>,  uid: <username>  passwd: <tunnel-cookie-value>  (2)200 OK, response with http session cookie is sent to the VIA client.  (2)CGI module receives and informs AUTH with username and tunnel cookie, which sends a radius authentication request  (3) Radius response with Rolename (which is the VIA client user-id used during login process).  CPPM |

NOTE: AUTH module in Controller sends Radius request once in SECTION-4.2 and the other in SECTION 4.4. CPPM will return the same Role name during both the occasions.

### Example configuration on controller:

aaa authentication via auth-profile "VIA-PASSWD"

default-role "mcd-passwd-role"

server-group "mcd-rad-serv"

download-role

aaa authentication via auth-profile "VIA-SSO"

default-role "mcd-sso-role"

server-group " mcd-cppm-sso"

sso-enable

download-role

aaa authentication via connection-profile “myconn”

server addr “10.3.18.5” internal-ip 10.3.18.5 desc “via-server”

auth-profile “VIA-SSO”

auth-profile “VIA-PASSWD”

aaa authentication via web-auth "default"

auth-profile "VIA-SSO" position 1

auth-profile “VIA-PASSWD” position 2

In the current solution “client user-id” will be returned as role-name. This Client user-id is considered to be unique across users (This is same as VIA client user-id that the user is using during login). Since this unique role-name is dynamic, controller may not have this role configured and hence will assign “default-role” configured in the corresponding auth-profile chosen by the VIA client.

### Need for tunnel-cookie token

SAML standards mention about SAML response to be used by the client to access the services. In the current proposal, VIA client (in addition to SAML response from IDP) will be provided with tunnel-cookie by the CPPM which is derived from SAML response. The reason for this being:

* SAML response validity could be really small (around 30 secs) and hence delay in IKE/IPSEC or latencies would result in SAML response validation failure during IKE/IPSEC tunnel establishment.
* Size of SAML response could be in KBs and adds to the message overhead.
* Controller need not be SAML-aware

As an alternate design VIA client could provide the SAML response instead of the above tunnel-cookie and controller could verify the hash of the SAML response without knowing the internals of SAML assertion response.

## VIA PROFILE DOWNLOAD

Using the above VIA tunnel-cookie, VIA client sends a HTTP GET request to /via/config to download VIA profile. Controller CGI module will request AUTH module with the rolename and AUTH replies with IKE/IPSEC configuration in the VIA AUTH/connection profile corresponding to the role. The response from controller to VIA client will be 200 OK with an XML body.

|  |
| --- |
| (1)VIA login https request to <controller>/via/config  (2)200 OK, response with connection profile is sent to the VIA client.  VIA client  Controller |

## VIA IKE/IPSEC CONNECTION

VIA client to controller IKE/IPSEC tunnel will be established using IKEv1 mode. The flow of packets is as indicated below.

Note: IKEv2 does not support downloadable role feature for VIA. Need investigation to support SSO with IKEv2 EAP authentication methods.

|  |
| --- |
| (2)XAUTH with tunnel cookie  VIA client  Controller  CPPM  (1)IKE Main mode negotiation  (3) Validation at controller for tunnel cookie. Initiates radius authorization request for username. If validation of tunnel cookie fails, sends back failure and VIA client needs to initiate VIA login process again to /via/sso  (4) Radius response with role name & Store-ids for which the user has access  (6) ACLs for the role downloaded from ACL Server which is applied to the inner VIA user  (5 a)XAUTH, mode config and Quick mode negotiation to bring up the IPSec tunnel. Data traffic can now flow through.  External ACL Server  (5b) HTTPS request to ACL Server with Rolename & store-ids for ACL download |

VIA client will initiate IKEv1 once it has the tunnel-cookie and provides the username/ tunnel-cookie as XAUTH\_USER\_NAME/XAUTH\_USER\_PASSWORD. IKE daemon on the controller will send XAUTH\_STATUS =1/0 to VIA client on successful/failure in authentication.

As indicated in Section – 4.2, CPPM will be saving the tunnel-cookie for around 600 secs. Any attempt to establish IKE/IPSEC tunnel within that time period, VIA user will not be requested for credentials again. Also, once the IKE/IPSEC tunnel is successfully established any subsequent IKE/IPSEC rekey VIA client user will not be prompted for credentials. Note, after the above initial 600 secs any new IKE/IPSEC tunnel establishment attempt will be prompted for VIA client credentials (as the tunnel-cookie will not be valid).

### Need for External ACL Server

AT&T/MCD have their own way to derive the ACL/ACE for a given VIA user. User is associated to store-ids and store-ids to subnets, from which we could derive the ACL/ACE. This ACL/ACE derivation is customer specific. So, the controller will download these ACLs from the ACL-servers (instead of CPPM).

### Radius Request attributes during XAUTH

In this proposal, the semantics of store number to subnet mapping and deriving corresponding relevant ACL/ACEs is handled by an external HTTPS server – hereinafter referred to as the ‘External ACL Server’

The External ACL server will interface with ATT’s existing Oracle DB that holds Store Number to Subnet mapping.

In the XAUTH RADIUS Access-Accept response, CPPM will send down VSAs as below.

NOTE: Existing Aruba-CPPM-Role will not be used, as AT&T wants VIA client username to be used as Role name instead of CPPM generating the rolename.

|  |  |
| --- | --- |
| **VSA** | **Meaning** |
| Aruba-User-Role | The dynamic role name to use by the controller, and in this case will be user-id |
| Aruba-External-Acl-Server-Query-Info | This will be the list of store numbers, returned in the same string format as received by the CPPM as part of the SAML Assertion response |

### Query for downloadable role definition

Controller uses curl command as under to query CPPM or External ACL server

|  |
| --- |
| curl -X POST –user <username>:<password> \    https:/10.3.18.5/<url \_*configured\_on*\_controller\_to\_query\_att\_acl\_server> \    -H 'accept: application/xml' \    -H 'content-type: application/x-www-form-urlencoded' \    -d ‘role=ikev1psk&query\_info=123%2C432%2C12' |

Today curl command used to download the ACL/ACE from CPPM does not verify the CPPM server certificate. But, AT&T wants verification of ACL server to be introduced. Once they provide the complete details of their ACL server certificate needs, appropriate changes will be made.

In the above Curl command, there are 3 variables external-server-fqdn, external-server-URI and username/password. New CLIs explained in SECTION-5.2 will be implemented.

### External ACL server Response formatC

The External ACL Server should process the request as below

* Take the value of ‘query\_info’ parameter from the HTTPS POST request – which is a list of store numbers.
* Use this to fetch the corresponding subnets from the ATT Oracle DB (say 10.134.22.0 and 10.22.135.0 as in the below example)
* Convert these subnets to ACEs as below

|  |
| --- |
| ip access-list session AccessACL  any network 10.134.22.0 255.255.255.0 any permit  any network 10.22.135.0 255.255.255.0 any permit |

* Respond to the HTTPS query with this ACL in the XML format given below

|  |
| --- |
| <?xml version="1.0" ?>  <CPPMAPIResponse xmlns="http://www.arubanetworks.com/cppm/apidefs/arubaroledefs/1.0">  <CPPMAPIHeader exportTime="2018-03-20 06:44:48 +000" version="1.0"/>  <StatusCode>  Success  </StatusCode>  <CPPMDownloadableRoles>  <CPPMDownloadableRole roleName="<role-name>" version="1">  <![CDATA[<role-name>  ip access-list session AccessACL  any network 10.134.22.0 255.255.255.0 any permit  any network 10.22.135.0 255.255.255.0 any permit  !  user-role “<role-name>”  access-list session AccessACL  !]]> </CPPMDownloadableRole>  </CPPMDownloadableRoles>  </CPPMAPIResponse> |

## ROLE DERIVATION

Controller downloads VIA client user-role and ACL/ACE from external ACL server. These user-role and ACL/ACE needs to be in the same format as existing downloadable-derived-role feature supported for VIA client users.

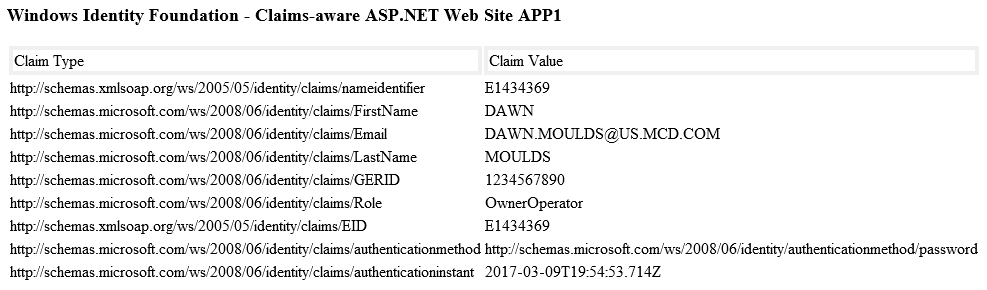
Since the number of users could be a lot and to avoid creating large number of roles on the controller, there is a need to define both static and dynamic role-names.

* Customer could define static roles on the controller when it is possible to reuse the roles for multiple users.
* CPPM could generate dynamic role-names for unique users. These dynamic roles will be deleted when the VIA client logs off.

### Sample McDonald Role specification

Below are the User definitions as shared by the customer.

|  |  |
| --- | --- |
| **MCD Role Description** | **value** |
| **Super User – Engineering Team** | admin |
| **Enterprise user – Networks Ops Team** | enterprise\_user |
| **Support Team/Help Desk - AtoS, McD Support Teams** | support |
| **Enterprise Reporting – Leadership, Digital** | report |
| **O/O – Owner/Operator** | owner\_operator |
| **OTP 2/3** | OTP2   or OTP3 |
| **McOpCo** | MCOPCO |



Generic Table:

|  |  |
| --- | --- |
| **SAML Attribute Name required** | **Attribute Value description** |
| **NameIdentifier** | Username (MCID/EID) |
| **FirstName** | Authenticated user’s first name |
| **LastName** | Authenticated user’s last name |
| **Email** | Authenticated user’s email |
| **EID** | Authenticated user’s EID |
| **Phone** | Authenticated user’s phone |
| **GERID** | Group Entity Ownership ID |
| **StoreList** | List of stores [will be 5 digit numbers] |
| **SpecialAccess**  (This is an optional attribute) | For **Phase I**, the value of attribute can only be ALL (if there’s a value)  For **Phase II**, there may be additional possible keywords for other types of special access (ex. “MCOPCO” “RMHC”, etc.). There may be more than one special keyword per user |
| **Role** | admin, enterprise\_user, support, report, owner\_operator, OTP2, OTP3, MCOPCO |

# Controller-side Configuration

The following new parameters will be added to AOS VIA auth-profile configuration as part of this feature.

|  |
| --- |
| aruba-controller) (VIA Authentication Profile "via\_sso\_profile") #?  auth-protocol           Authentication Protocol to be used.Default is PAP.  cert-cn-lookup          Check certificate common name against AAA server.                          Default is enabled.  client-cert-enable      Enable Client-certificate based authentication for VIA                          Profile download. Default is disabled.  clone                   Copy data from another VIA Authentication Profile  default-role            Assign default role  desc                    Description  download-role           Download Role from CPPM if not defined  max-authentication-fa.. Maximum auth failures before user is blacklisted.                          Range: 1-10. Default: 0.  no                      Delete Command  pan-integration         Require IP mapping at Palo Alto Networks firewalls  radius-accounting       Configure server group for radius accounting  rfc-3576-server         Configure RFC 3576 server  server-group            Name of server group  sso-acl-server-config   Configuration of SSO ACL server  sso-acl-server-host     IP address/Hostname of ACL server  sso-enable              Enable SSO based authentication for VIA Profile                          download. Default is disabled.  sso-redirect            Configuration of SSO redirect server |

## Enable SSO for VIA

The option will enable SSO way of authentication.

## CPPM SSO redirect URL

As in Section-4.2 VIA client sends its initial request to /via/sso on the controller and controller will redirect to CPPM configured in the corresponding Auth profile. The redirect URL on CPPM needs to be configured, by default request will be sent to the port-443.

## External ACL Server configuration

Controller will download the ACL/ACE corresponding to the site-ids from the external ACL server. Details like username/password, FQDN or IP and URI/port needs to be configured.

# Administration

## CLI

Display commands that will added

|  |
| --- |
| VIA Authentication Profile "mysos"  ----------------------------------  Parameter                                                         Value  ---------                                                         -----  Default Role                                                      myvia  Server Group                                                      myrad  Check certificate common name against AAA server    Enabled  Client-certificate based authentication for VIA Profile download  Disabled  SSO based authentication for VIA Profile download       Enabled  SSO redirect URI and port                                         /via/sso  433  SSO ACL server host                                               10.1.4.179  SSO ACL server configuration                                      test/\*\*\*\*\*\*\*\*  /xml  443  Authentication protocol                                           pap  Download Role from CPPM                                           Enable |

## UI

CLIs added will automatically add the UI fields

# Licensing

PEF-VPN license is required on the controller acting as the VPN concentrator for the VIA users.

For AOS Release 8.2 onwards per user VIA license: LIC-VIA will be required on controller.

# Migration Strategy

This feature will require CPPM to have an externally reachable IPaddress as the VIA client’s https requests will be redirected to CPPM.

# Interoperability

# Monitoring

There are existing IKE AMON feeds that should cover the tunnel related events. VIA specific changes need to be updated.

# Performance and scale

As explained in the Section, there are multiple redirect response/response during SSO login.

1. There is a need to measure the total time it takes for a single VIA client to be able to establish the tunnel and to be able to communicate with the controller and the N/W behind.
2. Performance numbers in terms of number of VIA clients that could be connected per sec.

# Unit tests and Guidelines for QA

* Check if SSO auth profile is configurable and is displayed in the corresponding “show command”. --done
* Check if SSO auth profile is shown in the web-browser along with other auth profiles---done
* Check if username/password credential feature works aswell with corresponding auth profile configured.
* Redirect
* Check if redirect URL for SSO is configurable and is displayed in the corresponding “show command”
* Check for max length of redirect URL
* SSO redirect works when SSO is enabled in the auth profile chosen
* SSO redirect should not work when SSO is not enabled in the auth profile chosen
* SSO redirect is sent to only to the active CPPM.
* Parse redirect
* SSO redirect should initiate a request back to controller for parsing the cookie and username and controller should return the same.
* VIA client should be able to login with the configured username and cookie on the browser
* Once login is successful, VIA client should be able to download the profile
* Once login is successful, VIA client should be able to establish VPN connection. Check for user with default role.
* Check XAUTH failure is sent when the password does not match.
* Check ACL server configuration is successful and the same is displayed with the corresponding show command.
* Check ACL is downloaded from the ACL server and VIA client is assigned appropriate role.
* Check client is assigned with the appropriate role if the role is already configured on the controller.
* All the above tests to be checked for opusone

# Open Issues

1. Should Controller/CPPM be the SP
   * Flow diagram was studied and discussed at length across teams. Team agreed on having CPPM as the SP.
   * STATUS: Closed
2. Can CPPM have public-ip and deployed in DMZ
   * Customer-facing team confirmed with AT&T that CPPM will have the public-ip and could be deployed in DMZ for VIA client to interact directly with CPPM
   * STATUS: Closed
3. Parsing of MCDonald defined assertion and converting to Aruba-defined Role and ACL/ACE.
   * CPPM team to come up with a proposal
   * STATUS: Closed
4. Can CPPM provide the tunnel-cookie and save it for 600secs. This will allow VIA clients to attempt to download the profile or establish IKE/IPSEC tunnel during that time.
   * CPPM team to confirm
   * STATUS: Closed
5. Is there a need to send the tunnel-cookie as Vendor ID payload during IKE/IPSEC negotiation?
   * AUTH and VIA team to discuss
   * STATUS: Closed
6. The roles returned in Section-4.3 and Section-4.4 from CPPM are the same?
   * AUTH team to investigate the current working behavior
   * STATUS: Closed
7. ATT/MCD needs to provide External ACL Server to confirm the functionality of ACL download
   * PLM to confirm, otherwise could delay the feature delivery
   * STATUS: Open
8. ATT/MCD needs to provide the details of the IDP used to confirm the functionality
   * PLM to confirm, otherwise could delay the feature delivery
   * STATUS: Open
9. ACL/ACE download for a given role from CPPM or from external SSL server needs to be confirmed
   * Sriram to confirm with the customer
   * STATUS: Closed
10. Should CPPM query ORACLE-DB to download the subnets corresponding to site-id
    * Sriram to confirm with the customer
    * STATUS: Closed

# Deferred Features

NA

# Dependencies

Third party IDP used by McDonald’s

# Meeting Minutes

* Add the need for VIA license for 8.2 and above
* Add details for tunnel-cookie generation in CPPM
* To add details about VIA using WPF
* To add details about via default role configuration on controller

# Appendix