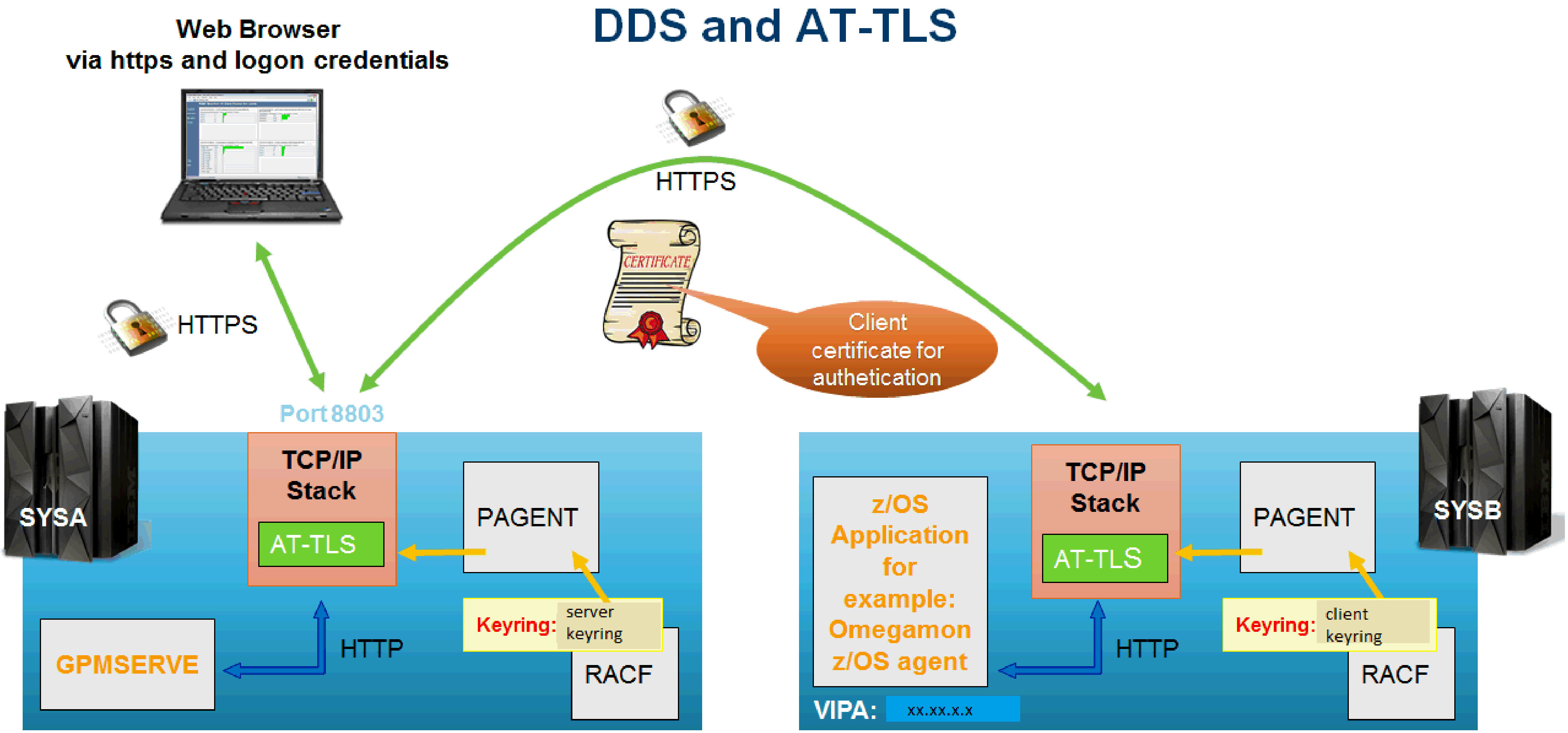
RMF development has provided an AT-TLS setup that would allow to use a protected userid for the started task running the OMEGAMON z/OS agent.

With this AT-TLS setup we have the following:
The OMEGAMON agent would use a client certificate for authentication. In addition, the connection to DDS would be secured by use of https instead of http. With the authentication via a client certificate, you would need to specify the IP address of the Omegamon z/OS agent in the http_noauth parameter of RMF DDS (GPMSERVE) so that the DDS will not do any further authentication. This allows you to use a protected userid for the OMEGAMON started task.

Other programs (like web browsers) that will retrieve DDS data can use https requests and logon credentials (userid/password or passticket)

Here is a picture that will help to understand the DDS with AT-TLS setup, followed by the AT-TLS definitions.
(ip addresses and keyring names would be edited) :



Here is an AT-TLS policy example for this AT-TLS DDS setup:

Rules for AT-TLS for the server (left side picture) :
#Ruleset for application, running with VIPA xx.xx.x.x

```
TTLSRule                        DDSServerRule
{
  LocalPortRange                 8803
  Jobname                        GPMSERVE
  Direction                      Inbound
  RemoteAddr                     xx.xx.x.x ->for example application
                                 specific dynamic vipa of omegamon z/OS
TTLSGroupActionRef               DDSServerGRP
  TTLSEnvironmentActionRef       DDSServerENV
  Priority                       2
}
TTLSGroupAction                  DDSServerGRP
{
  TTLSEnabled                    On
  TRACE                          255
}
TTLSEnvironmentAction            DDSServerENV
{
  HandshakeRole                  ServerWithClientAuth
  TTLSKeyringParms
  {
     Keyring                     serverkeyring
  }
}
```

Rules for AT-TLS for the client (ride side picture):
#RMF Distributed Data Server Client Rule

```
TTLSRule                         DDSClientRule
{
  RemotePortRange                8803
  RemoteAddr                     z.zzz.zz.zz ->IP address of DDS
  Direction                      Outbound
  TTLSGroupActionRef             DDSClientGRP
  TTLSEnvironmentActionRef       DDSClientENV
}
TTLSGroupAction                  DDSClientGRP
{
  TTLSEnabled                    On
  TRACE                          255
}
TTLSEnvironmentAction            DDSClientENV
{
  HandshakeRole                  Client
  TTLSKeyringParms
  {
    Keyring                      clientkeyring
  }
}
```

Rules for other external access (example, web browser) :
#Ruleset for external access via https and logon credentials (for example via web browsers)

```
TTLSRule                          DDSServerRuleBrowser
{
  LocalPortRange                  8803
  Jobname                         GPMSERVE
  Direction                       Inbound
  TTLSGroupActionRef              DDSServerBrowserGRP
  TTLSEnvironmentActionRef        DDSServerBrowserENV
  Priority                        1
}
TTLSGroupAction                   DDSServerBrowserGRP
{
  TTLSEnabled                     On
  TRACE                           255
}
TTLSEnvironmentAction             DDSServerBrowserENV
{
  HandshakeRole                   Server
  TTLSKeyringParms
  {
     Keyring                      serverkeyring
  }
}
```

The example AT-TLS policy provides the following:

DDS server rule for programs (like OMEGAMON z/OS agent) , running with a specific IP address, which need to do authentication with a client certificate (DDSServerRule with HandshakeRole ServerWithClientAuth)
Client rule for programs (like OMEGAMON z/OS agent), contacting the DDS and provide a client certificate (DDSClientRule with HandshakeRoleClient)
DDS server rule for other access, e.g. browser, which only talk https to the DDS (DDSServerRuleBrowser with HandshakeRole Server)

RACF Keyring serverkeyring
DDS Server keyring;
server certificate (default), enables to talk HTTPS certificates of clients which are allowed to talk to the DDS

RACF Keyring clientkeyring
DDS Client keyring
client certificate for authentication (default) server certificate, to verify that the server certificate is trusted