GOPAS

Web Application Proxy

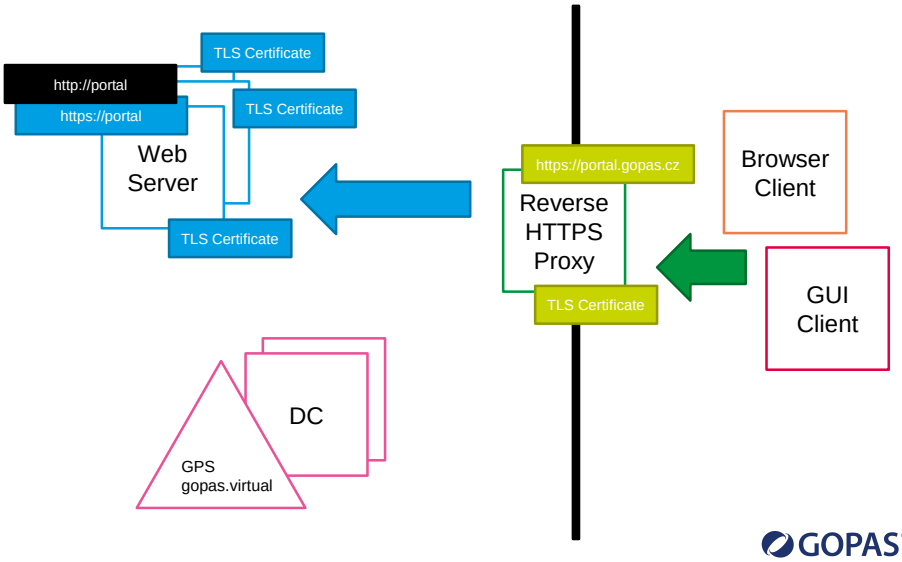
Ing. Ondřej Ševeček | GOPAS a.s. |
MCSM:Directory2012 | MCM:Directory2008 | MVP:Enterprise Security | CEH: Certified Ethical Hacker | CHFI: Computer Hacking Forensic Investigator |
ondrej@sevecek.com | www.sevecek.com |

Motivation

- TMG discontinued
  - TCP/IP/ICMP/IPSec/etc. inspection fully replaced with Windows Firewall
  - intrusion prevention filters included in Windows Defender and Microsoft Security Essentials
  - problematic expansion of reverse HTTPS publishing
- Secure reverse HTTPS publishing
  - Windows authentication at network perimeter
  - Forms-based (cookie) authentication with non-browser fallback to Basic and/or persistent cookie

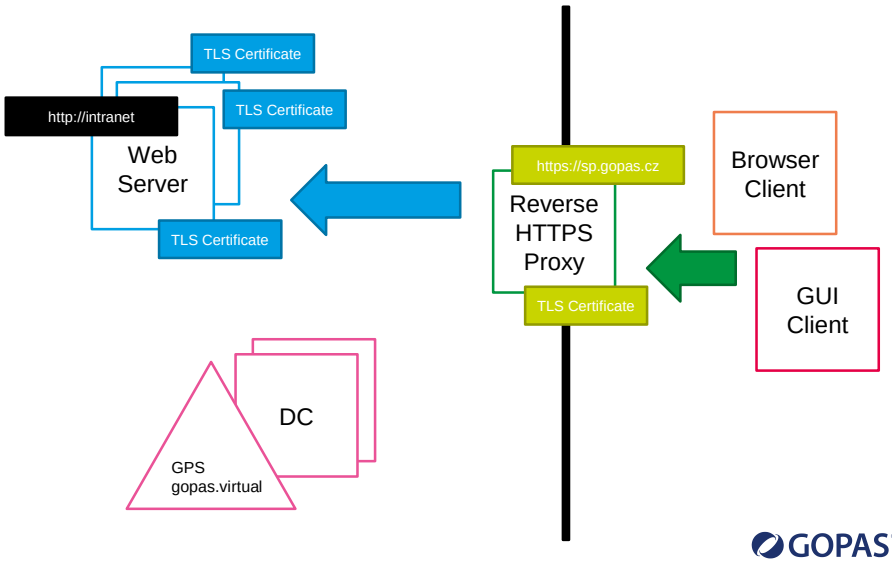
GOPAS

Principal scenario (internal HTTP or HTTPS)



GOPAS

Principal scenario (SharePoint, AAM)



GOPAS

Another bit of motivation

- SharePoint
- not everything requires authentication
- HTTP level protocol exploits
  - many many many IIS modules to pass

GOPAS

Reverse HTTPS proxy general requirements

- Require HTTPS from client
  - possibly redirect to secure traffic
  - rather do not redirect to discourage HTTPS strip
  - minimize number of public TLS certificates
- Decrypt HTTPS at the perimeter
  - possibly inspect, define rules or extend with third-party
  - translate external URI to internal host names and paths
  - forward different host header
- Authenticate users at the perimeter
  - Windows authentication against Active Directory
  - allow other authentication databases if necessary
- Forward user credentials to the application
  - Windows authentication (WIA) delegation with Kerberos
  - claims with Windows Identity Foundation

GOPAS

WAP on Windows 2012 R2

- Require HTTPS from client
  - possibly redirect to secure traffic
  - rather do not redirect to discourage HTTPS strip
  - minimize number of public TLS certificates
- Decrypt HTTPS at the perimeter
  - possibly inspect, define rules or extend with third-party
  - translate external URI to internal host names and paths
  - forward different host header
- Authenticate users at the perimeter
  - Windows authentication against Active Directory
  - allow other authentication databases if necessary
- Forward user credentials to the application
  - Windows authentication delegation with Kerberos
  - claims with Windows Identity Foundation
- TLS SNI as a bonus over TMG
  - plus Extended Protection for Authentication (NTLM mutual authenitcation)

GOPAS

Wait. First make Kerberos work internally

- Web server GPS-WFE1
- Web application accessible at http://portal
- Application pool running under ApplicationPoolIdentity
- IIS Windows Authentication enabled, Kernel Mode Authentication enabled
- DNS name portal.gopas.virtual = A

- Set servicePrincipalName (SETSPN) on GPS-WFE1
  - http/portal
  - http/portal.gopas.virtual

GOPAS

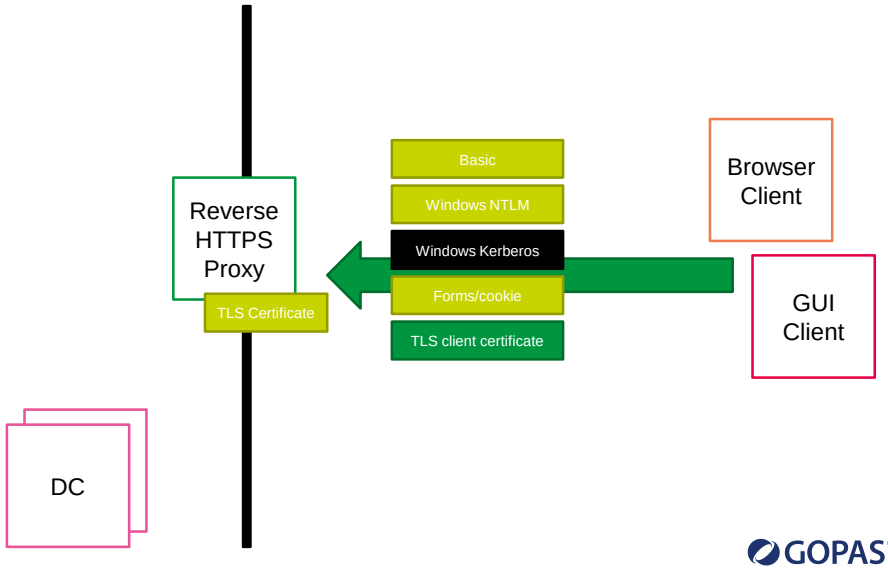
Wait some more. Yet make Kerberos work internally even for SharePoint

- Web server GPS-SP
- Web application accessible at http://intranet
- Application pool running under sp-intranet-web
- IIS Windows Authentication enabled, Kernel Mode Authentication disabled
- DNS name intrnaet.gopas.virtual = A

- Set servicePrincipalName (SETSPN) on sp-intranet-web
  - http/intranet
  - http/intranet.gopas.virtual

GOPAS

External authentication

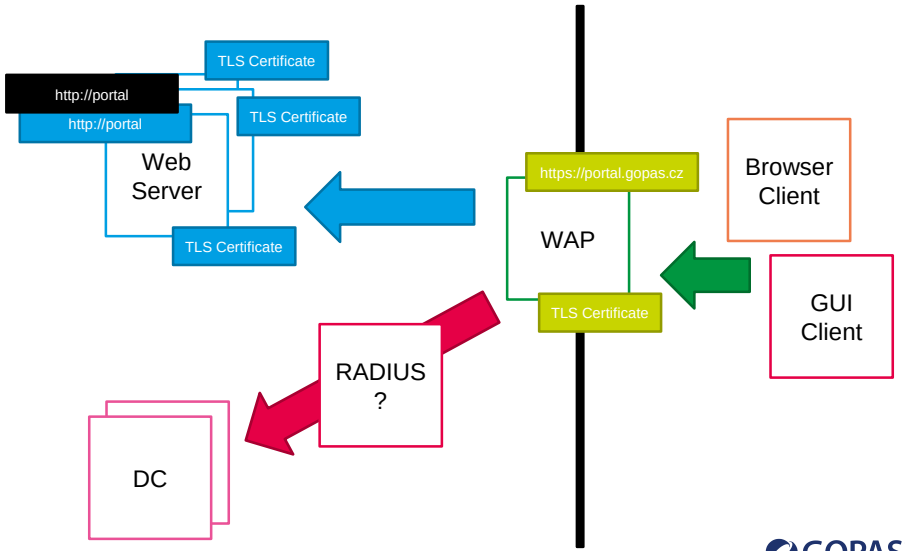


GOPAS

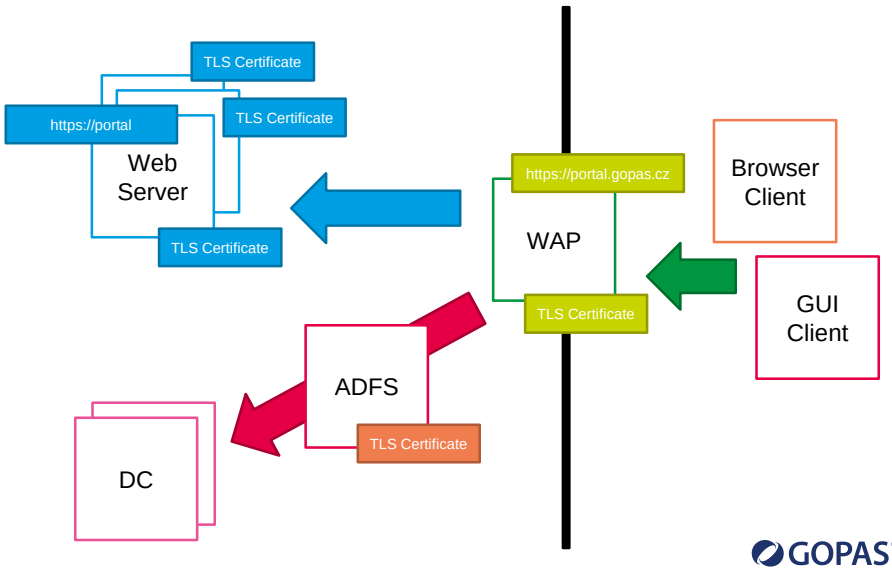
External authentication challenges

External authentication	Facts	Internal forwarding	Notes
Basic	plain-text TLS encrypted no SSO	easy	no browser sign-out no timeout non-browser clients
Windows NTLM	SSO	Kerberos constrained delegation	complicated sensitive
Windows Kerberos	not possible without direct contact with DC	Kerberos constrained delegation	impossible
Forms/cookie	plain-text no SSO session vs. persistent cookie	easy claims SAML token	sign-out timeout browser clients
TLS client certificate	safe against password guessing safe against HTTP exploits	Kerberos constrained delegation claims SAML token	only for "partners" can use smart-cards both clients

Scenario with an authentication server



Scenario with ADFS authentication server



GOPAS

Standard web-based authentication

- Active Directory Federation Services (ADFS)
- HTTP server providing several web based authentication mechanisms
  - Active Directory (ADDS)
  - Active Directory Lightweight Directory Services (ADLDS)
  - any third party
- Produces claims or cookies in various formats
  - WS-Trust or SAML-Token for active clients
  - WS-Federation (SAML 1.1) and SAML 2.0 for passive clients
  - OAuth for semi-passive clients
- Required by Office365/AzureAD for on-premises hybrid deployments

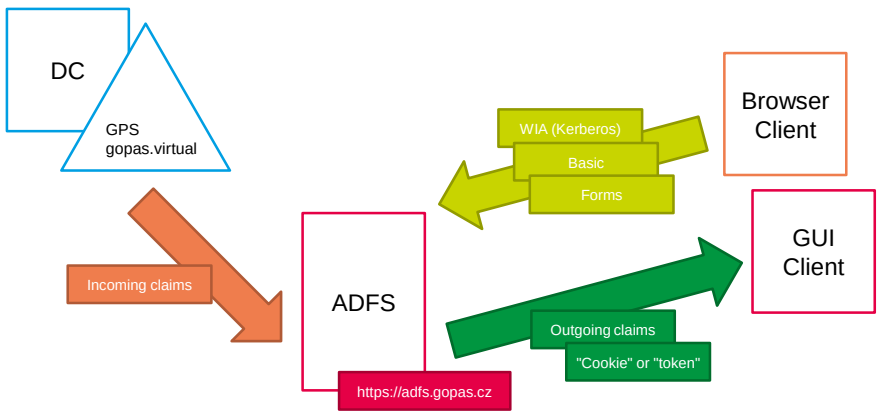
GOPAS

ADFS version history

Version	OS	Notes
ADFS 1.0	Windows 2003 R2	included runs in IIS
ADFS 1.1	Windows 2008 Windows 2008 R2	included runs in IIS
ADFS 2.0	Windows 2008 Windows 2008 R2	download runs in IIS
ADFS 2.1	Windows 2012	included runs in IIS
ADFS 3.0	Windows 2012 R2	included direct hosting on HTTP.SYS TLS SNI support PowerShell only config (plus HTML)

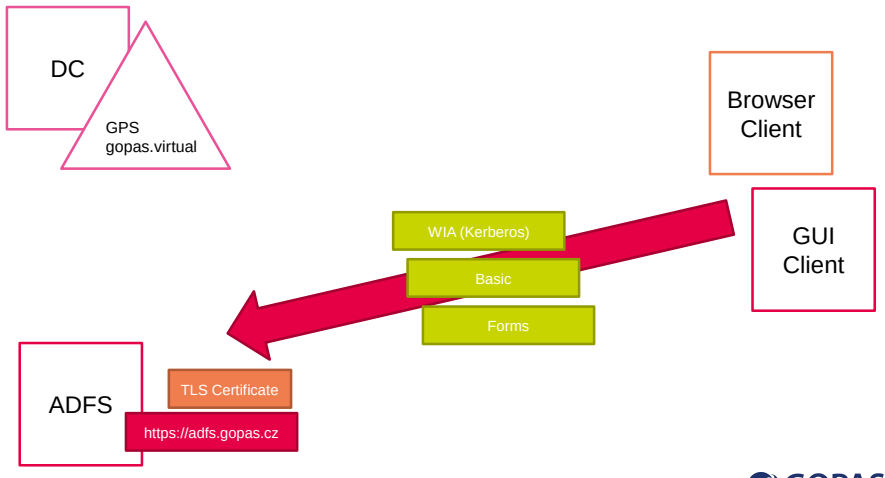
GOPAS

Simple ADFS terminology



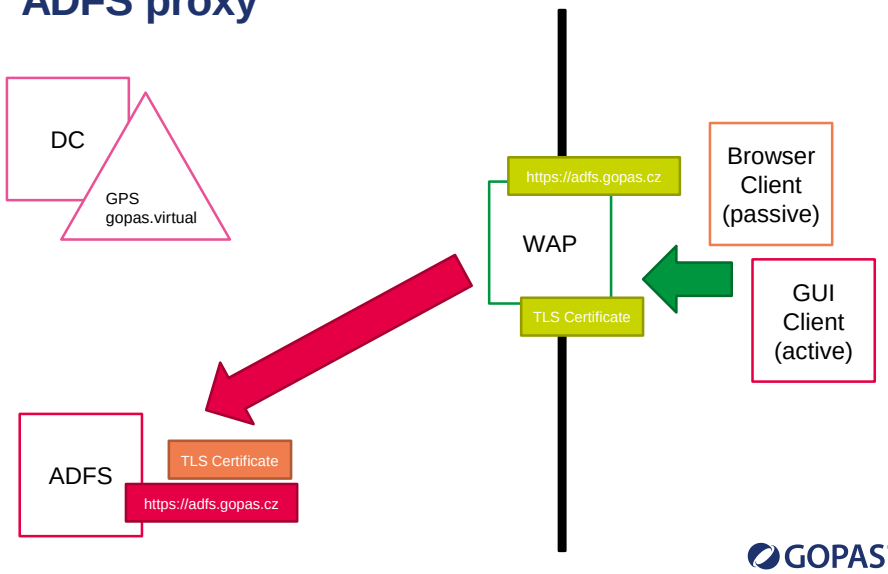
GOPAS

ADFS internal testing



GOPAS

ADFS public access with WAP acting as an ADFS proxy



GOPAS

ADFS configuration notes

- Must be Domain Admins member to install ADFS
  - some stupid customer requirement
- Installer account must be sysadmin in DB
- ADFS service account gets servicePrincipalName
  - Domain Admins can write it, does not require self registration
- Creates and AD container
  - CN=Program Data,CN=Microsoft,CN=ADFS,CN=CertificateSharingContainer,DC=x
- NETSH HTTP SHOW SSLCERT
- NETSH HTTP SHOW SERVICESTATE | findstr :443
- WAP connects over Admin$ to ADFS
- ADFS service account must be member of WAAG if user attributes are to be used as filters on incoming claims

GOPAS

Testing ADFS from browser

- F12 developer toolbar
  - does not show authentication headers
- Fiddler with TLS inspection

GOPAS

Testing ADFS from browser

- https://adfs.gopas.cz/federationmetadata/2007-06/federationmetadata.xml
  - anonymously available
- https://adfs.gopas.cz/adfs/ls/idpinitiatedsignon.htm
  - manually initiated from browser
- https://adfs.gopas.cz/adfs/ls?wsignin1.0&wtrealm=https://portal.gopas.cz
  - WS-Federation sign-in URL, you receive SAML1.1 token
  - configured as: WS-Federation Passive Endpoints on the Endpoints tab
- https://adfs.gopas.cz/adfs/ls?SAMLRequest=Base64request
  - SAML2.0 sign-in URL, returns SAML2.0 token
  - configured as: SAML Assertion Consumer Endpoints on the Endpoints tab
- https://adfs.gopas.cz/adfs/oauth2/authorize?response_type=code&client_id=11111111-2222-3333-4444-123456789012&redirect_uri=https://portal.gopas.cz&resource=https://portal.gopas.cz
  - OAuth sign-in URL, returns OAuth token, only for active clients
  - configured as: no endpoint plus use Get-AdfsClient and Add-AdfsClient
- sign-out
  - https://adfs.gopas.cz/adfs/ls/?wa=wsignout1.0
  - https://adfs.gopas.cz/adfs/ls/?wa=wsignout1.0&wreply=https://www.google.cz

GOPAS

Testing ADFS from browser

- Get-AdfsProperties
- requires extended protection for WIA
  - to enable WIA for FireFox set ExtendedProtectionTokenCheck = 'None'
  - type 'about:config', filter for 'ntlm', add 'adfs.gopas.cz' to 'network.automatic-ntlm-auth.trusted-uris' setting
- WIASupportedUserAgents
  - MSIE, MSAuthHost/1.0/In-Domain, Trident/7.0, MSIPC, Windows Rights Management Client

GOPAS

HTTP cookies generally

- Name=Value; Name=Value; ...
- Path=/subPath
  - limited to a subpath
- Domain=.gopas.cz
  - can enable cookie from a subdomain to go to other thirt-level subdomains
- Expires=23-May-2015 22:13:08 GMT
- Max-Age=[seconds]
  - expirations in browser are not enforced
  - servers expire cookies themselves

GOPAS

How ADFS knows what is internal and what is an external client

- ADFS proxy must forward requests with x-ms-proxy and x-ms-endpoint-absolute-path
  - you cannot simply proxy internal WAP-ADFS communication with Fiddler, because it is mutually authenticated
- Any reverse web proxy supported, not just WAP

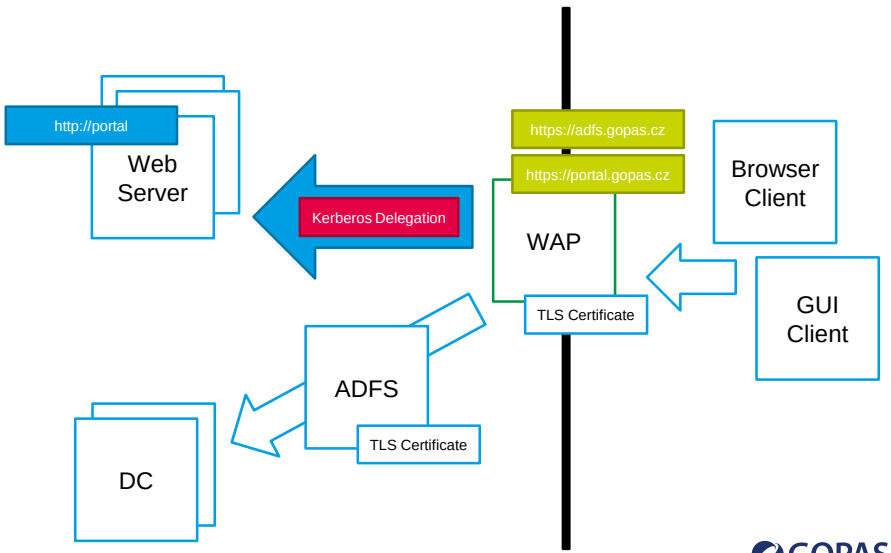
GOPAS

Testing ADFS from GUI client

- use Fiddler to decrypt HTTPS
- use Windows Identity Foundation to request active responses
  - cannot produce SAML 2.0 (SAML-Protocol) cookie based responses

GOPAS

Publishing simple WIA web application



GOPAS

Kerberos delegation requirements

- Kerberos working internally WAP-WEB
  - http/portal
  - http/portal.gopas.virtual
  - or any arbitrary SPN specified in the WAP configuration
- Kerberos delegation for WAP server
  - Trust this computer to specified services only, Use any authentication protocol
  - WAP member of Windows Authorization Access Group (WAAG)
  - restart WAP machine

GOPAS

Alternative attribute stores

- LDAP connection string
  - LDAP://localhost:11111/cn=Users,o=GOPAS
  - ADFS authenticates against ADLDS with its service account
- SQL connection string
  - Server=GPS-DATA;Database=PartnerAccounts;Integrated Security=True;Encrypt=True

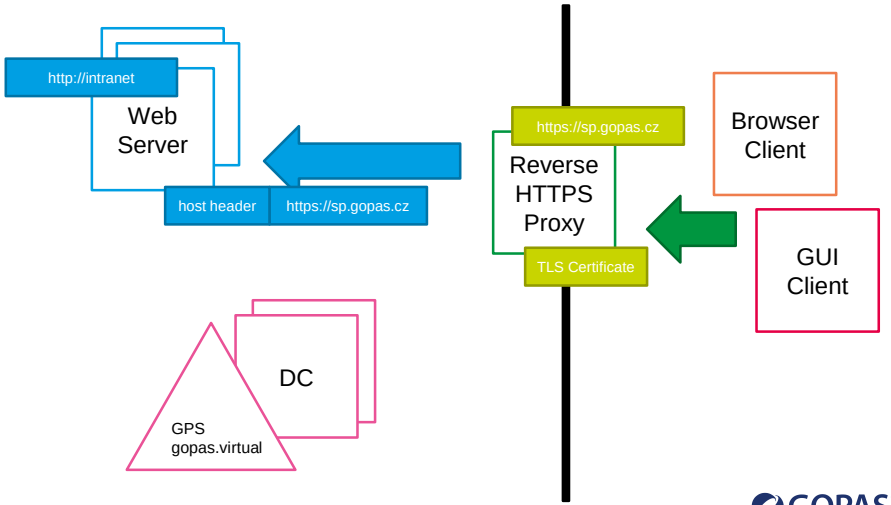
GOPAS

Publishing SharePoint

- Best practice to run internal SP web on public name since the very start
  - SharePoint must know the host name that client uses
- Running SharePoint on internal name
  - WAP should always forward with the external host header
  - WAP cannot define different host header for a different internal name/IP translation
  - WAP must use HOSTS or internal DNS records

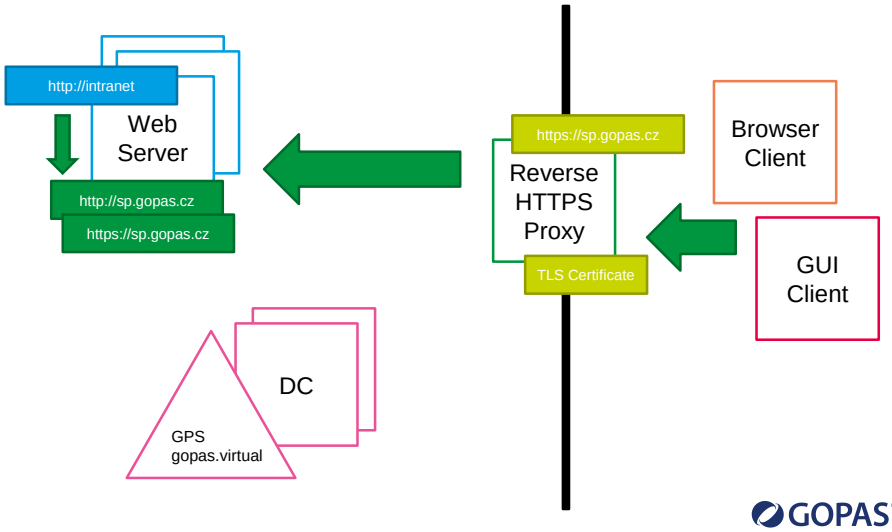
GOPAS

Scenario for SharePoint publishing ok if non-host header web binding or the same public/private host header (maybe AAM)



GOPAS

Extend web application first (maybe AAM) for host header web binding



GOPAS

Thank you

- My training in GOPAS

- GOC166 - Advanced ADFS
- GOC167 - Troubleshooting Remote Access, VPN and DirectAccess
- GOC169 - ISO/IEC 2700x in Windows environment
- GOC171 - Active Directory Troubleshooting
- GOC172 - Kerberos Troubleshooting
- GOC173 - Enterprise PKI Deployment
- GOC175 - Advanced Windows Security
- GOC174 - SharePoint Troubleshooting

GOPAS