



A study on Windows authentication & Prox-Ez

THCon 2023

Geoffrey Bertoli & Pierre Milioni

Who are we?

2



- **Geoffrey Bertoli (@YofBalibump) & Pierre Milioni (@b1two_)**
 - Pentesters at Synacktiv
- **Working for Synacktiv**
 - Offensive security
 - ~140 ninjas: pentest, reverse engineering, development, CSIRT
 - 4 locations: Paris, Rennes, Lyon, Toulouse, (soon at Lille) & remote
 - We are hiring! → apply@synacktiv.com

Introduction



■ A little bit of history

- NTLM introduced in 1993 with Windows NT 3.1
- NTLMv2 since Windows NT 4.0 SP4 – 1998
- But here comes the mighty Kerberos
 - Became a standard in 1993 (v5)
 - Introduced in Windows 2000
- NTLM still widely used nowadays



■ Multiple mitigations against relay

- NTLMv1 → v2 (not our focus today)
- NTLM – MIC (not our focus today)
- NTLM EPA (Extended Protection for Authentication)
 - Channel Binding
 - Service Binding
- Kerberos
 - Whole new authentication mechanism
 - More complex than NTLM
 - Mutual authentication
 - Fix relay attack

Introduction



■ Still of interest today

- Lack of (proper) documentation of some topics
- Not supported by all tools
- Lack of tooling for these authentications over HTTP

■ Prox-Ez

- MitM proxy for Windows authentication over HTTP(s)
- Single file, born to be patched

Agenda



- **Quick overview of NTLM**
- **NTLM and relaying**
- **NTLM-EPA (Extended Protection for Authentication)**
 - Channel Binding
 - Service Binding
- **What about Kerberos?**
 - Over HTTP
 - Security overview



■ **New Technology Lan Manager**

- Windows authentication protocol
- Single Sign-On
- Based on challenge/response exchange
- **Authenticates a session** (TCP connection in case of HTTP)
 - May cause issues/slowdowns with programs that creates new TCP connections for each request

BurpSuite now supports TCP connection reuse

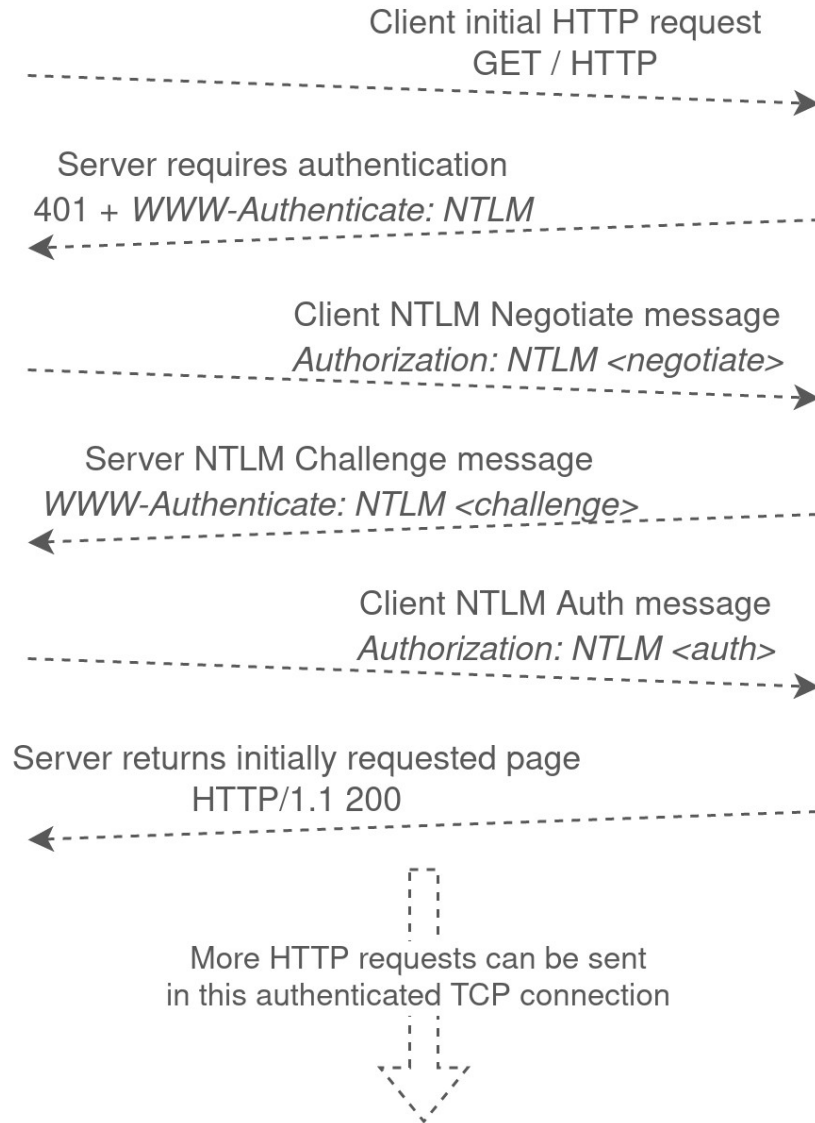
■ ... over HTTP



Client's
browser



Web server

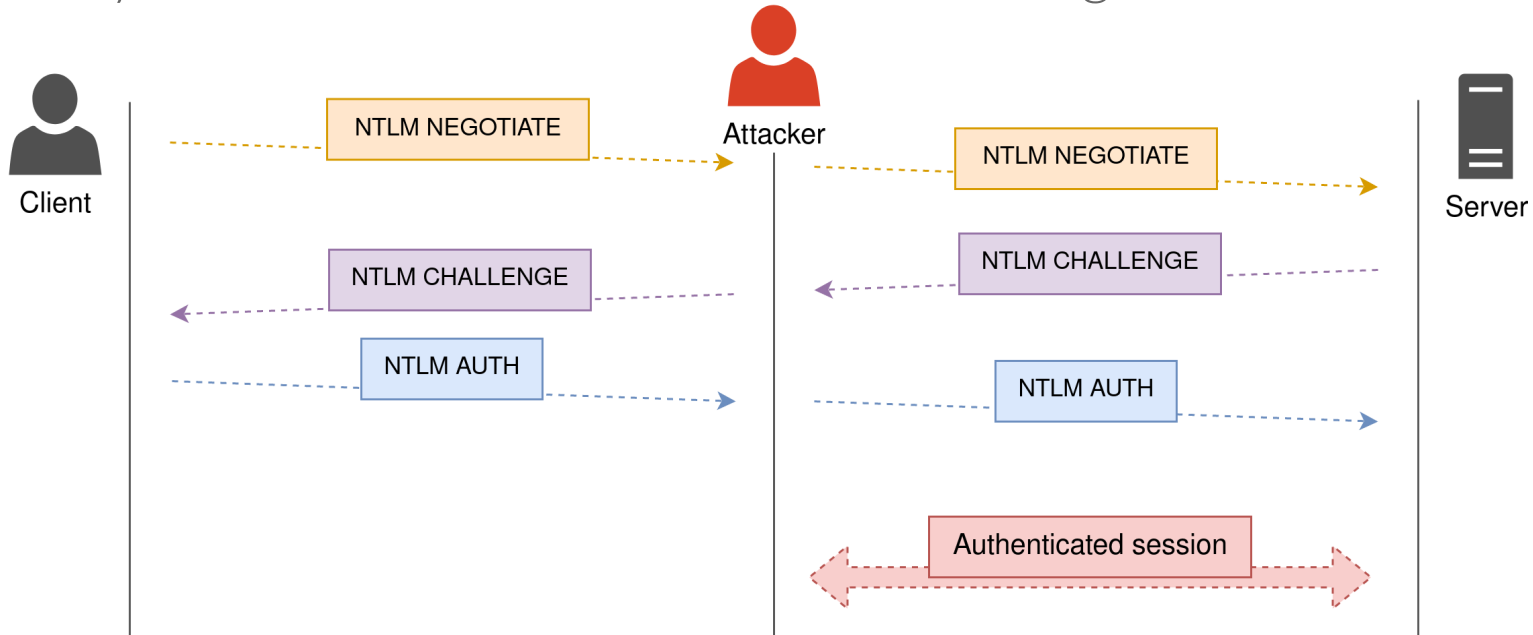


NTLM



■ NTLM relaying

- Attacker in a relaying position (able to forward messages from a client)
- Relays the client's authentication to the targeted server

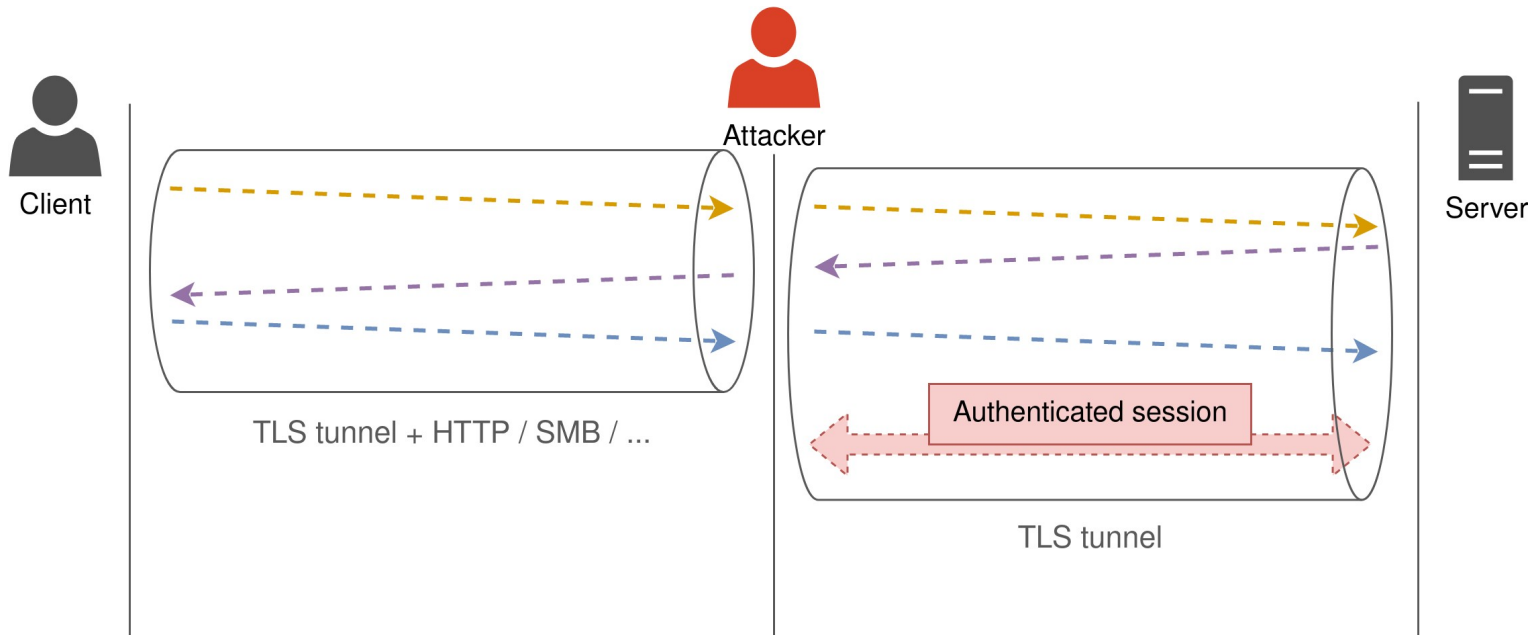


NTLM



■ NTLM relaying – Over TLS

- Attacker in a relaying position
- Relays the client's authentication to the targeted server

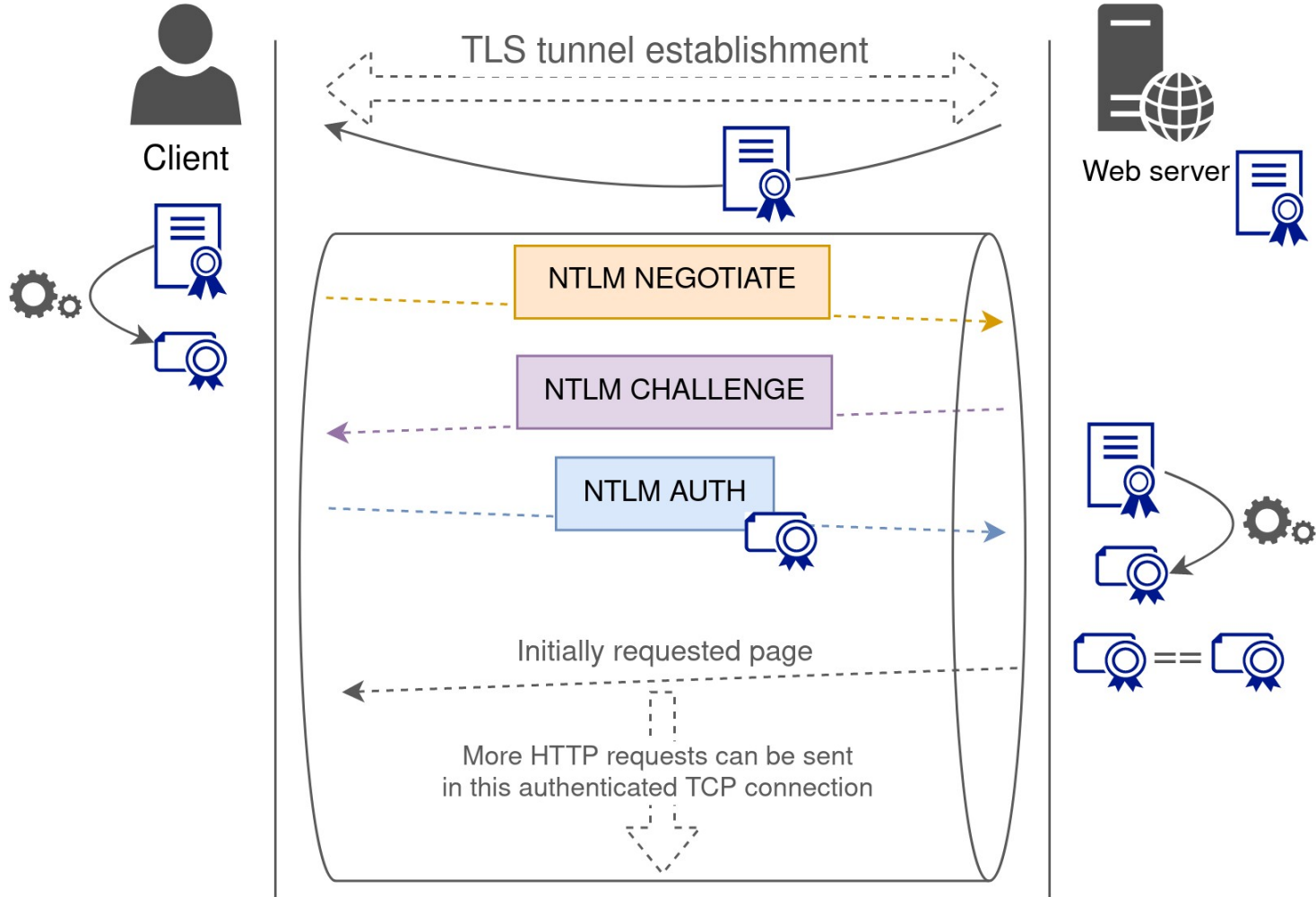




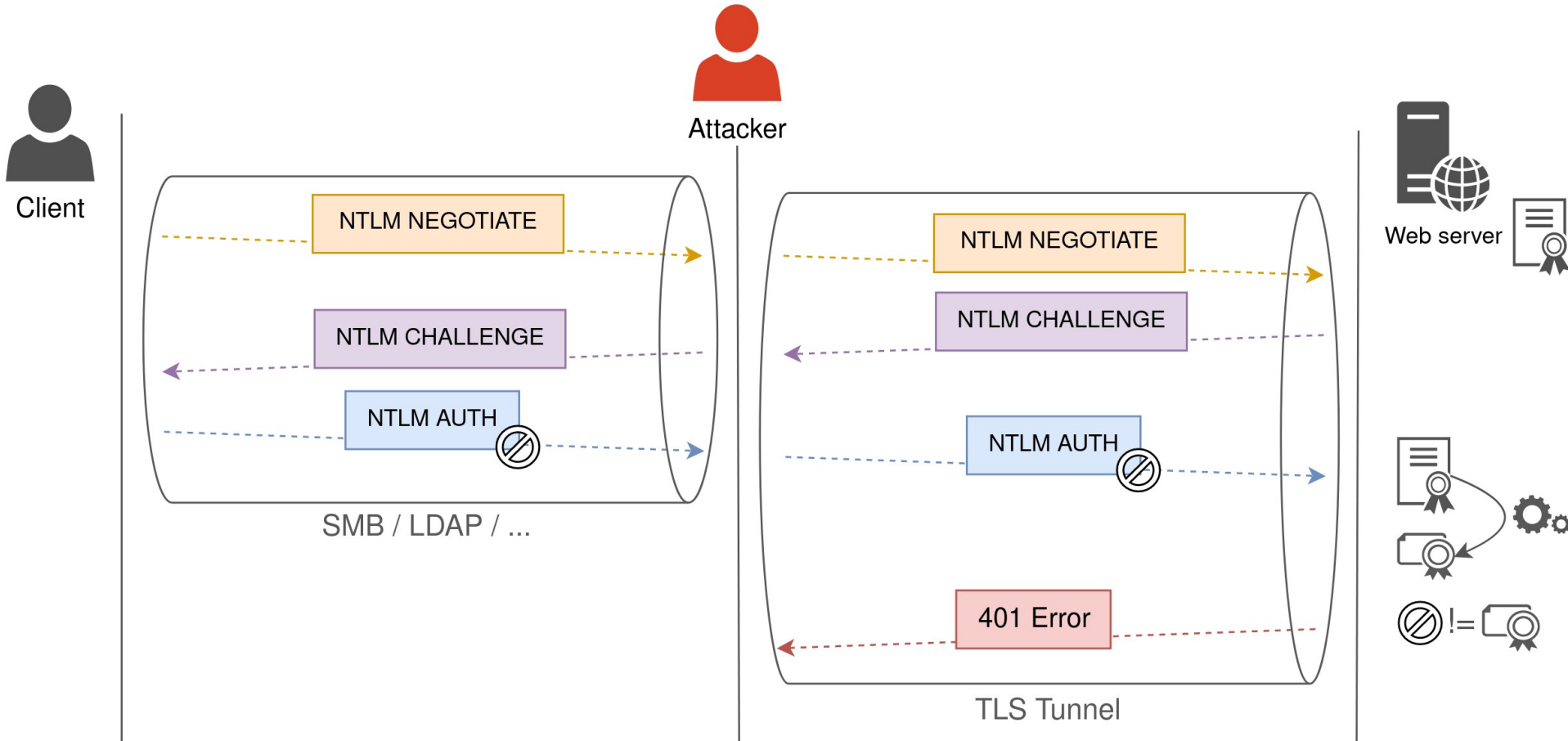
■ Channel Binding

- Microsoft's solution to protect against MitM attacks
- Used on TLS based communications
- “Binds” the authentication to the outer TLS channel
 - Adds a token that depends on the TLS tunnel into the NTLM authentication
- Can be *required* by the server
 - Any client without the proper token are denied access

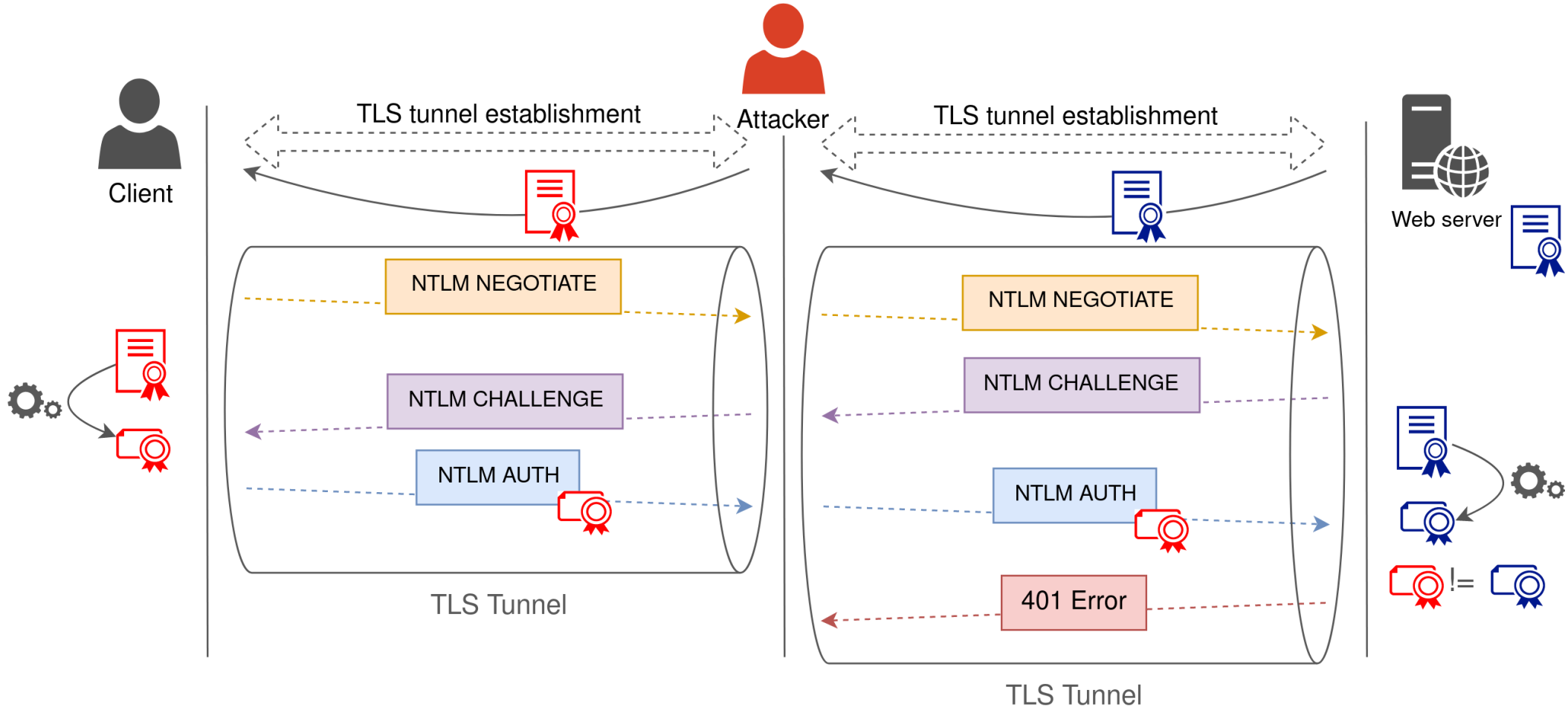
NTLM EPA – Channel Binding



NTLM EPA – Channel Binding



NTLM EPA – Channel Binding



NTLM EPA – Channel Binding



■ **CBT: Channel Binding Token**

- Hash of the server's certificate
- With the hash function used to compute the certificate's signature

Certificate signature's hash function	MD5 / SHA-1	Other hash function	No hash function / multiple hash functions
CBT's hash function	SHA-256	Signature's hash function	Undefined

NTLM EPA – Channel Binding

■ Channel Bindings attribute

- Derived from the CBT
- Inserted in the NTLM *AUTH* message
- Cannot be modified
 - Protected by the MIC value (HMAC)
 - protected by the *Flags* attribute
 - protected by *NTProofStr*
 - protected by the client's secret



```
GET / HTTP/1.1\r\n
Host: lab.mylab\r\n
Connection: keep-alive\r\n
Cache-Control: max-age=0\r\n
[truncated]Authorization: NTLM T1RMTVNTUADAAAAGAAAYfGAAAAEAQQBcAAAAAAAAAB0AQACAAIAHQBA...
  NTLM Secure Service Provider
    NTLMSSP identifier: NTLMSSP
    NTLM Message Type: NTLMSSP_AUTH (0x00000003)
    Lan Manager Response: 0000000000000000000000000000000000000000000000000000000000000000
    LMv2 Client Challenge: 0000000000000000
    NTLM Response: f4759ae9f42a05e0205987f244a0e96f0101000000000000d473c5e28b37d7014e8aec...
      Length: 260
      Maxlen: 260
      Offset: 112
    NTLMv2 Response: f4759ae9f42a05e0205987f244a0e96f0101000000000000d473c5e28b37d7014e8aec...
      NTProofStr: f4759ae9f42a05e0205987f244a0e96f
      Response Version: 1
      Hi Response Version: 1
      Z: 000000000000
      Time: Apr 22, 2021 15:26:37.529186000 UTC
      NTLMv2 Client Challenge: 4e8aec4f6d936e28
      Z: 00000000
      Attribute: NetBIOS domain name: DESKTOP-OUBJGUL
      Attribute: NetBIOS computer name: DESKTOP-OUBJGUL
      Attribute: DNS domain name: DESKTOP-OUBJGUL
      Attribute: DNS computer name: DESKTOP-OUBJGUL
      Attribute: Timestamp
      Attribute: Flags
      Attribute: Channel Bindings
        NTLMV2 Response Item Type: Channel Bindings (0x000a)
        NTLMV2 Response Item Length: 16
        Channel Bindings: 45f5a466fcdf1531df8e70317a5ee4d3
      Attribute: Target Name: HTTP/1.1/lab.mylab
      Attribute: End of list
      Z: 00000000
    Domain name: NULL
    User name: user
    Host name: Wololo
```


NTLM EPA – Channel Binding



■ Channel Binding

- Still not supported by many clients
→ no authentication possible if EPA is required
- How to use our tools against EPA protected websites?

MitM Proxy – Prox-EZ (“prox easy”)



■ Why?

- Be able to use any tool against HTTP(s) servers using
 - NTLM
 - NTLM-EPA
 - *Kerberos*
- Be able to control the authentication

MitM Proxy – Prox-EZ (“prox easy”)



■ How?

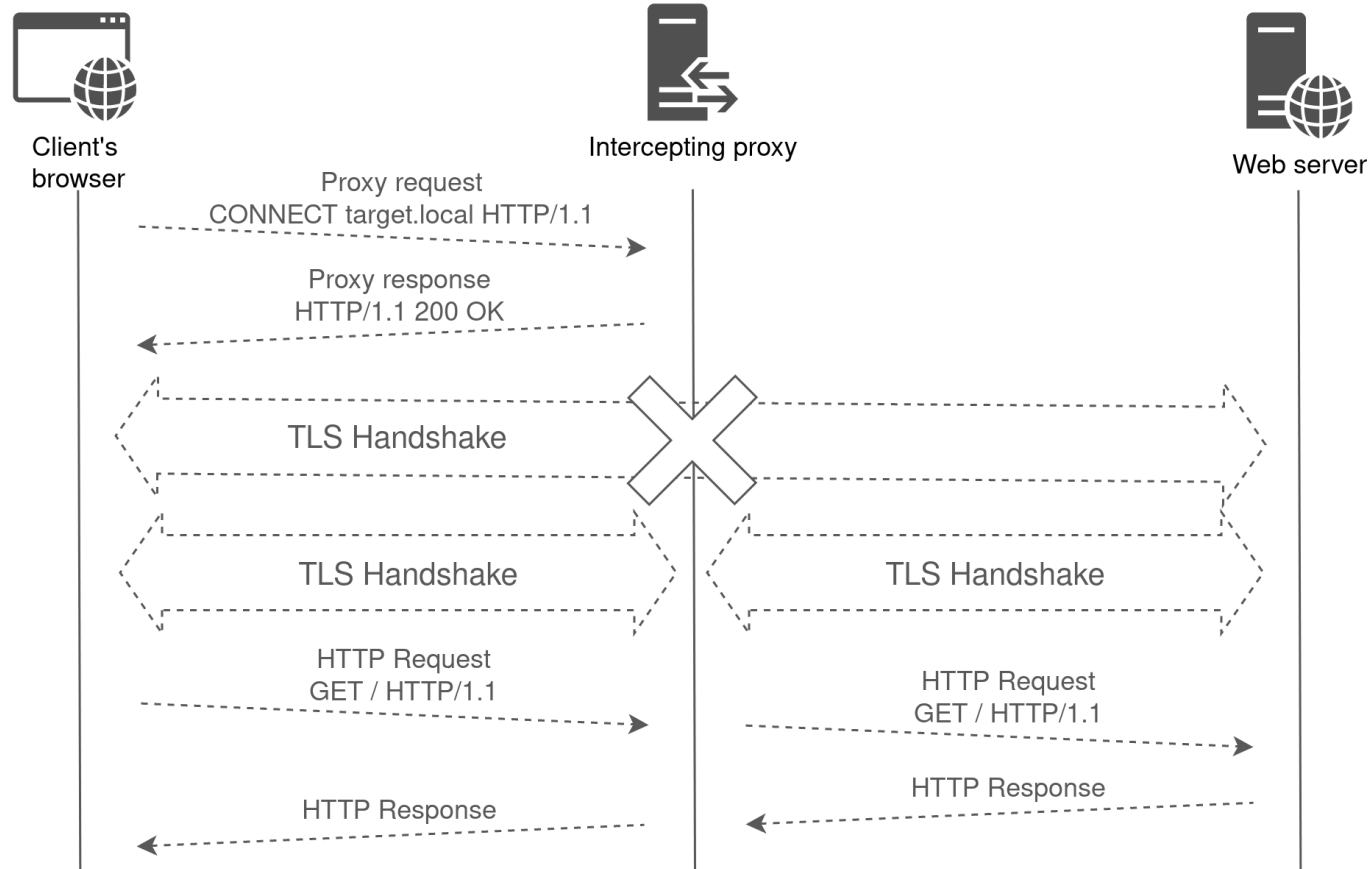
- Has to work with TLS
 - TLS interception
 - Register a custom certificate authority on the client
 - Generate on-the-fly certificates
- Good documentation on *mitmproxy* website

MitM Proxy – Prox-EZ (“prox easy”)



How?

- TLS interception

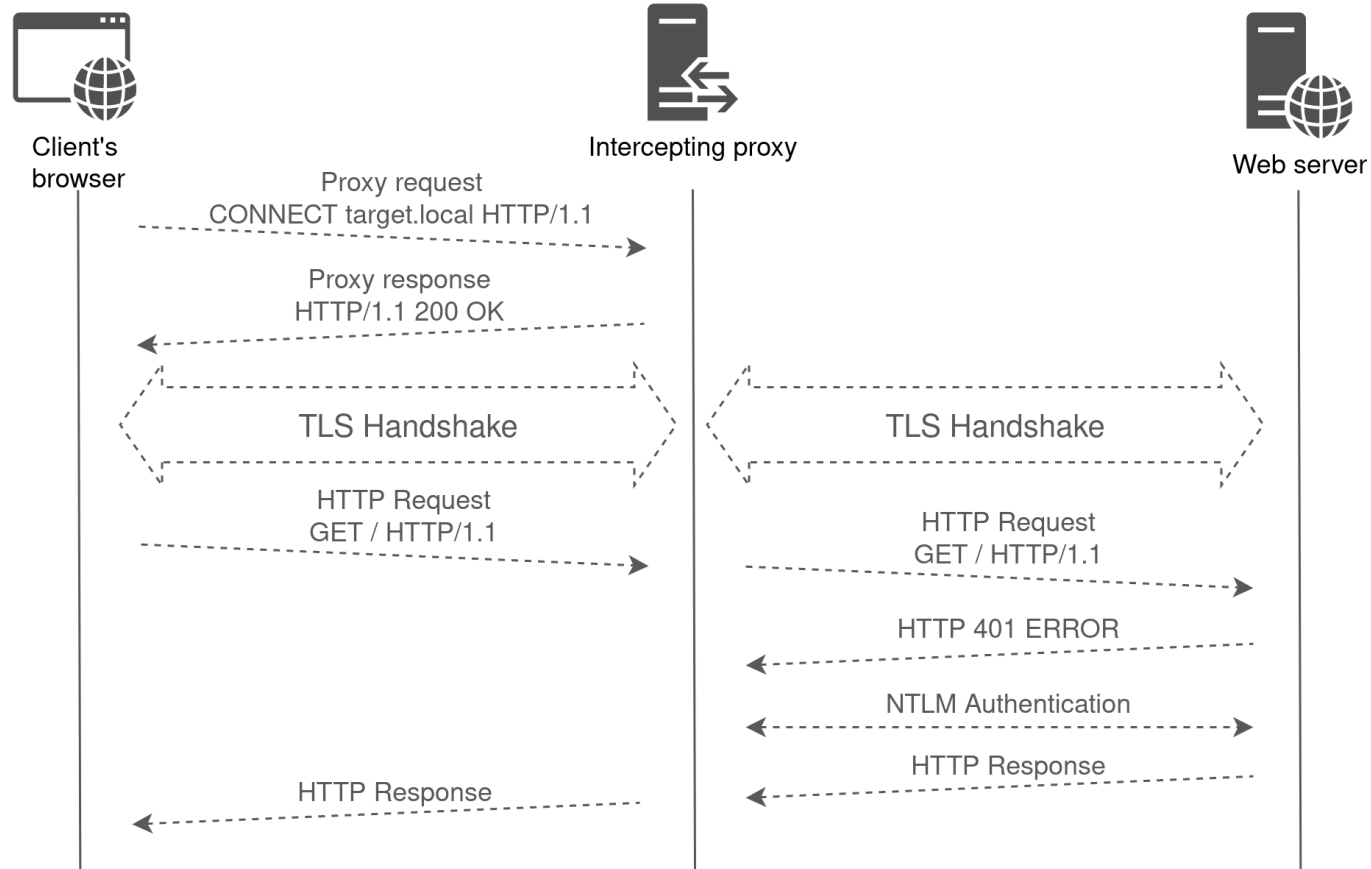


MitM Proxy – Prox-EZ (“prox easy”)



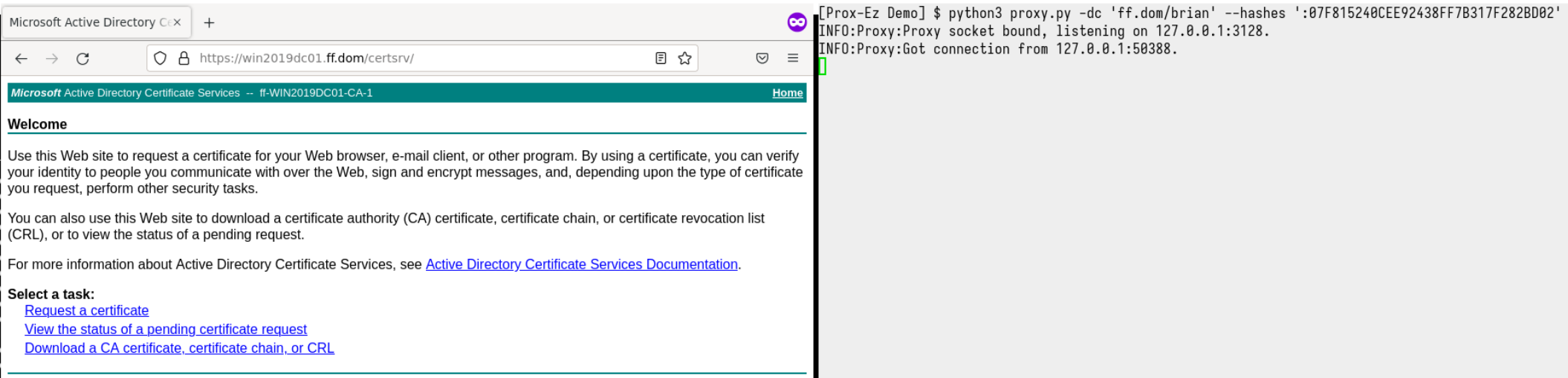
How?

- TLS interception



MitM Proxy – Prox-EZ (“prox easy”)

Demo



The image shows a split-screen view. On the left is a web browser window displaying the Microsoft Active Directory Certificate Services page for 'ff-WIN2019DC01-CA-1'. The page includes a 'Welcome' message and instructions on how to request a certificate, download a CA certificate, or view the status of a pending request. On the right is a terminal window showing the execution of the Prox-EZ proxy tool. The terminal output indicates that the proxy socket is bound to 127.0.0.1:3128 and that it has successfully established a connection from 127.0.0.1:50388.

```
[Prox-Ez Demo] $ python3 proxy.py -dc 'ff.dom/brian' --hashes ':07F815240CEE92438FF7B317F282BD02'  
INFO:Prox:Proxy socket bound, listening on 127.0.0.1:3128.  
INFO:Prox:Got connection from 127.0.0.1:50388.
```

MitM Proxy – Prox-EZ (“prox easy”)



Demo

The screenshot shows a browser window on the left and a terminal window on the right. The browser window displays a sign-in page for the domain `win2019dc01.ff.dom`. The page contains the text "This site is asking you to sign in." followed by input fields for "Username" and "Password". At the bottom of the form are "Cancel" and "Sign in" buttons. The terminal window on the right shows the following output:

```
[Prox-Ez Demo] $ python3 proxy.py -dc 'ff.dom/brian' --hashes ':07F815240CEE92438FF7B317F282BD02' --no-epa
INFO:Proxy:Proxy socket bound, listening on 127.0.0.1:3128.
INFO:Proxy:Got connection from 127.0.0.1:60828.
ERROR:Proxy:Error while performing authentication, stopping. Error details: Authentication failed, probably bad credentials or server does not support NTLM or bad EPA (chanel binding or service binding).
```



■ Service Binding

- Channel Binding requires TLS, now what about plain HTTP?
 - (Don't do plain HTTP)
- Microsoft implemented a new protection
 - Service Binding
- Same objective as Channel Binding → Prevent MitM attacks

NTLM EPA – Service Binding

■ Service Binding

- New attribute in the NTLM *AUTH* message
 - Identifies the targeted resource

709	3784.5819019...	10.137.0.61	10.137.0.47	HTTP	332	HTTP/1.1 401 Unauthorized , NTLMSSP_CHALLENGE (text/html)
710	3784.5819745...	10.137.0.47	10.137.0.61	TCP	56	47442 → 80 [ACK] Seq=1806 Ack=6198 Win=62848 Len=0
711	3784.5820976...	10.137.0.47	10.137.0.61	HTTP	1068	GET / HTTP/1.1 , NTLMSSP_AUTH, User: ff.dom\brian
712	3784.6008005...	10.137.0.61	10.137.0.47	TCP	56	80 → 47442 [ACK] Seq=6198 Ack=2818 Win=2102272 Len=0
713	3784.7048618...	10.137.0.61	10.137.0.47	HTTP	1006	HTTP/1.1 200 OK (text/html)
714	3784.7261469...	10.137.0.61	10.137.0.47	TCP	1006	[TCP Retransmission] 80 → 47442 [PSH, ACK] Seq=6198 Ack=2818
715	3784.7261754...	10.137.0.47	10.137.0.61	TCP	68	47442 → 80 [ACK] Seq=2818 Ack=7148 Win=64128 Len=0 SLE=6198
716	3784.7519562...	10.137.0.47	10.137.0.61	HTTP	471	GET /favicon.ico HTTP/1.1

Offset: 112

▼ NTLMv2 Response: 3ea30ac482a5dff7109d3c4b0b2bfa20101000000000000aaad3e9b46fd801cbb9072b...

NTProofStr: 3ea30ac482a5dff7109d3c4b0b2bfa2

Response Version: 1

Hi Response Version: 1

Z: 000000000000

Time: Nov 18, 2022 12:09:27.828829800 UTC

NTLMv2 Client Challenge: cbb9072bdf42fa55

Z: 00000000

- ▶ Attribute: NetBIOS domain name: FF
- ▶ Attribute: NetBIOS computer name: WIN2019SRV01
- ▶ Attribute: DNS domain name: ff.dom
- ▶ Attribute: DNS computer name: WIN2019SRV01.ff.dom
- ▶ Attribute: DNS tree name: ff.dom
- ▶ Attribute: Timestamp
- ▶ Attribute: Flags
- ▶ Attribute: Channel Bindings

▼ Attribute: Target Name: HTTP/win2019srv01.ff.dom
NTLMV2 Response Item Type: Target Name (0x0009)
NTLMV2 Response Item Length: 48
Target Name: HTTP/win2019srv01.ff.dom

▶ Attribute: End of list

NTLMV2 Response Item Type: End of list (0x0000)

NTLM EPA – Service Binding

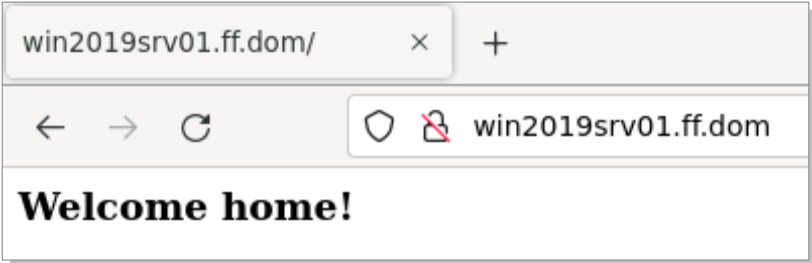


■ Service Binding

- New attribute in the NTLM *AUTH* message
 - Identifies the targeted resource
 - Taken from the browser URL

+	711	3784.5820976...	10.137.0.47	10.137.0.61	HTTP	1068	GET / HTTP/1.1 , NTLMSSP_AUTH, User: ff.dom\brian
-	712	3784.6008005...	10.137.0.61	10.137.0.47	TCP	56	80 → 47442 [ACK] Seq=6198 Ack=2818 Win=2102272 Len=0
-	713	3784.7048618...	10.137.0.61	10.137.0.47	HTTP	1006	HTTP/1.1 200 OK (text/html)
-	714	3784.7261469...	10.137.0.61	10.137.0.47	TCP	1006	[TCP Retransmission] 80 → 47442 [PSH, ACK] Seq=6198 Ack=2818
-	715	3784.7261754...	10.137.0.47	10.137.0.61	TCP	68	47442 → 80 [ACK] Seq=2818 Ack=7148 Win=64128 Len=0 SLE=6198
+	716	3784.7519562...	10.137.0.47	10.137.0.61	HTTP	471	GET /favicon.ico HTTP/1.1

```
Offset: 112
NTLMv2 Response: 3ea30ac482a5dff7109d3c4b0b2bfa20101000000000000aaad3e9b46fbd801cbb9072b...
  NTProofStr: 3ea30ac482a5dff7109d3c4b0b2bfa2
  Response Version: 1
  Hi Response Version: 1
  Z: 000000000000
  Time: Nov 18, 2022 12:09:27.828829800 UTC
  NTLMv2 Client Challenge: cbb9072bdf42fa55
  Z: 00000000
  Attribute: NetBIOS domain name: FF
  Attribute: NetBIOS computer name: WIN2019SRV01
  Attribute: DNS domain name: ff.dom
  Attribute: DNS computer name: WIN2019SRV01.ff.dom
  Attribute: DNS tree name: ff.dom
  Attribute: Timestamp
  Attribute: Flags
  Attribute: Channel Bindings
  Attribute: Target Name: HTTP/win2019srv01.ff.dom
    NTLMv2 Response Item Type: Target Name (0x0009)
    NTLMv2 Response Item Length: 48
    Target Name: HTTP/win2019srv01.ff.dom
  Attribute: End of list
  NTLMv2 Response Item Type: End of list (0x0000)
```





■ Service Binding

- New attribute in the NTLM *AUTH* message
 - Identifies the targeted resource
 - Taken from the browser URL
- If the authentication targets another server than the one receiving the authentication → denied access



■ Service Binding

- The web server needs to be configured with the proper SPNs
 - No implicit SPN
 - All the alternative DNS records
- Bad integration in IIS
 - No graphical option
 - Manual modification of `C:\Windows\System32\inetsrv\Config`



■ Service Binding

```
# C:\Windows\System32\inetsrv\Config
<location path="Default Web Site">
  <system.webServer>
    <security>
      <authentication>
        <windowsAuthentication enabled="true" useKernelMode="false">
          <providers>
            <clear />
            <add value="NTLM" />
          </providers>
          <extendedProtection tokenChecking="Require" flags="Proxy,ProxyCohosting">
            <spn name="HTTP/win2019srv01.ff.dom" />
          </extendedProtection>
        </windowsAuthentication>
        <anonymousAuthentication enabled="false" />
      </authentication>
    </security>
  </system.webServer>
</location>
```

NTLM EPA – Service Binding



■ Service Binding

<i>flags options</i>	Behavior	Remark
Empty / None	Only verify CBT	HTTP is not protected ; HTTPs is protected
<i>Proxy</i>	Only verify SPN	HTTP is not working (no authentication possible) ; HTTPs is protected
<i>Proxy,ProxyCohosting</i>	Only verify SPN	Both HTTP and HTTPs are protected and work
<i>Proxy,NoServiceNameCheck</i>	Does not verify anything	HTTP is not working (no authentication possible) ; HTTPs is not protected but a SPN has to be provided (any value)
<i>Proxy,ProxyCohosting,NoServiceNameCheck</i>	Does not verify anything	Both HTTP and HTTPs are not protected (no SPN required)



■ Service Binding

- Service Binding configuration is cumbersome
- Default EPA configuration → Service Binding not enforced
- Enforced EPA but plain HTTP available → vulnerable to MitM attacks

MitM Proxy – Prox-EZ (“prox easy”)



- **Even if not widely used**

- Prox-Ez implements EPA-Service binding

The screenshot shows a browser window with the address bar containing `https://win2019srv01.ff.dom`. The page content displays **Welcome home!**. To the right, a terminal window shows the following output:

```
[Prox-Ez Demo] $ python3 proxy.py -dc 'ff.dom/brian:mysuperpassword1!'
INFO:Proxy:Proxy socket bound, listening on 127.0.0.1:3128.
INFO:Proxy:Got connection from 127.0.0.1:45602.
```

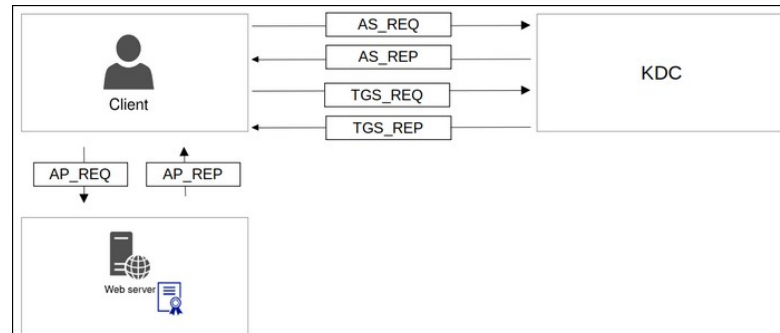
The screenshot shows a browser window with the address bar containing `https://win2019srv01.ff.dom`. The page content displays a sign-in form for `win2019srv01.ff.dom` with the text "This site is asking you to sign in." and a "Username" input field. To the right, a terminal window shows the following output:

```
[Prox-Ez Demo] $ python3 proxy.py -dc 'ff.dom/brian:mysuperpassword1!' --spn HTTP/test.com
INFO:Proxy:Proxy socket bound, listening on 127.0.0.1:3128.
INFO:Proxy:Got connection from 127.0.0.1:38226.
ERROR:Proxy:Error while performing authentication, stopping. Error details: Authentication failed, probably bad credentials or server does not support NTLM or bad EPA (chanel binding or service binding).
INFO:Proxy:Got connection from 127.0.0.1:38236.
```



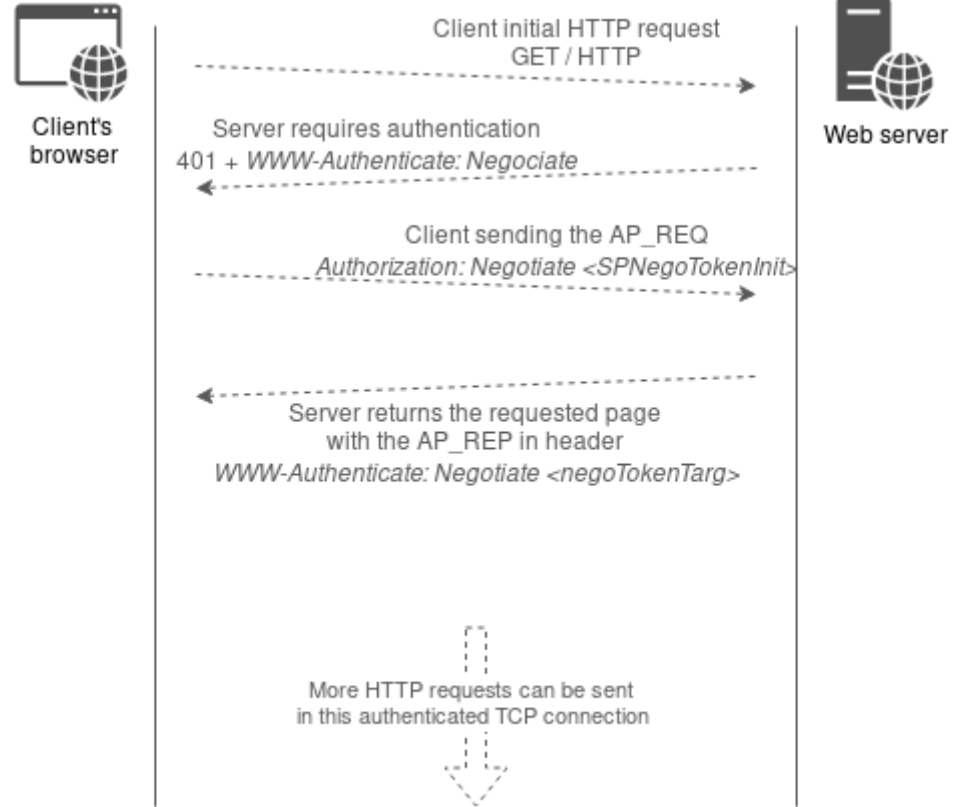

■ Why Kerberos ?

- Microsoft recommend enabling EPA as primary mitigation against relay attack (such as PetitPotam)
- In addition, disable NTLM and replace it by Kerberos
- Kerberos feature “Mutual Authentication”



■ ...over HTTP?

- Similar to NTLM
- The client sent the AP_REQ in a specific header





Let's have a closer look

```
HyperText Transfer Protocol
> GET / HTTP/1.1\r\n
Host: win-gc9km3m6ipp.domainest.local\r\n
Connection: keep-alive\r\n
Cache-control: max-age=0\r\n
[truncated]Authorization: Negotiate YIIHuwYGkwYBBQCoIHzCCB00gMDAuBgkqhkiC9xIBAgIGCSqGSIb3EgECAgYKKwYBBAGCNwICHgYKKwYBBAGCNwICCqKCw0EggcJYIIHBQYJK
655-APS-Generic-Security-Service-Application-Program-Interface
OID: 1.3.6.1.5.5.2 (SPNEGO - Simple Protected Negotiation)
Simple Protected Negotiation
  negTokenInit
    mechTypes: 4 items
    mechToken: 6082070506092a864886f71201020201006e8206f4308206...
    krb5_blob: 6082070506092a864886f71201020201006e8206f4308206...
      KRB5 OID: 1.2.840.113554.1.2.2 (KRB5 - Kerberos 5)
      krb5_tok_id: KRB5_AP_REQ (0x0001)
      Kerberos
        ap-req
          pvno: 5
          msg-type: krb-ap-req (14)
          padding: 0
          ap-options: 20000000 (mutual-required)
            ticket
              tkt-vno: 5
              realm: DOMAINTEST.LOCAL
                sname
                  name-type: kRB5-NT-SRV-INST (2)
                  sname-string: 2 items
                    SNameString: HTTP
                    SNameString: WIN-GC9KM3M6IPP.domainest.local
              enc-part
                authenticator
                  etype: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)
                  cipher: a2e320e3a8bfd7c5513a2665a329e0e2d56fdeb90bd99b13...
Upgrade-Insecure-Requests: 1\r\n
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36\r\n
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9\r\n
Accept-Encoding: gzip, deflate\r\n
Accept-Language: fr-FR,fr;q=0.9,en-US;q=0.8,en;q=0.7\r\n
If-None-Match: "bf5837cd4fd71:0"\r\n
If-Modified-Since: Wed, 15 Dec 2021 16:54:55 GMT\r\n
```

Kerberos



■ Let's have a closer look

```
> HTTP/1.1 304 Not Modified\r\n
Accept-Ranges: bytes\r\n
ETag: "bfb5837cd4f1d71:0"\r\n
Server: Microsoft-IIS/10.0\r\n
[truncated]WWW-Authenticate: Negotiate oYG1MIgyoAMKAQChCwYJKoZIGvcSAQICooGdBIGaYIGXBgkqkhiG9xIE
  GSS-API Generic Security Service Application Program Interface
    Simple Protected Negotiation
      negTokenTarg
        negResult: accept-completed (0)
        supportedMech: 1.2.840.48018.1.2.2 (MS KRB5 - Microsoft Kerberos 5)
        responseToken: 60819706092a864886f71201020202006f8187308184a003...
        krb5_blob: 60819706092a864886f71201020202006f8187308184a003...
          KRB5 OID: 1.2.840.113554.1.2.2 (KRB5 - Kerberos 5)
          krb5_tok_id: KRB5_AP_REP (0x0002)
        Kerberos
          ap-rep
            pvno: 5
            msg_type: krb_ap_rep (15)
            enc-part
              etype: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)
              cipher: 48ed6d1e0cbb92b30df00ba8b9432c0494d9cd1b1fe376e3...
Persistent-Auth: true\r\n
X-Powered-By: ASP.NET\r\n
Date: Tue, 22 Nov 2022 18:07:36 GMT\r\n
\r\n
[HTTP response 3/4]
[Time since request: 0.055348000 seconds]
```



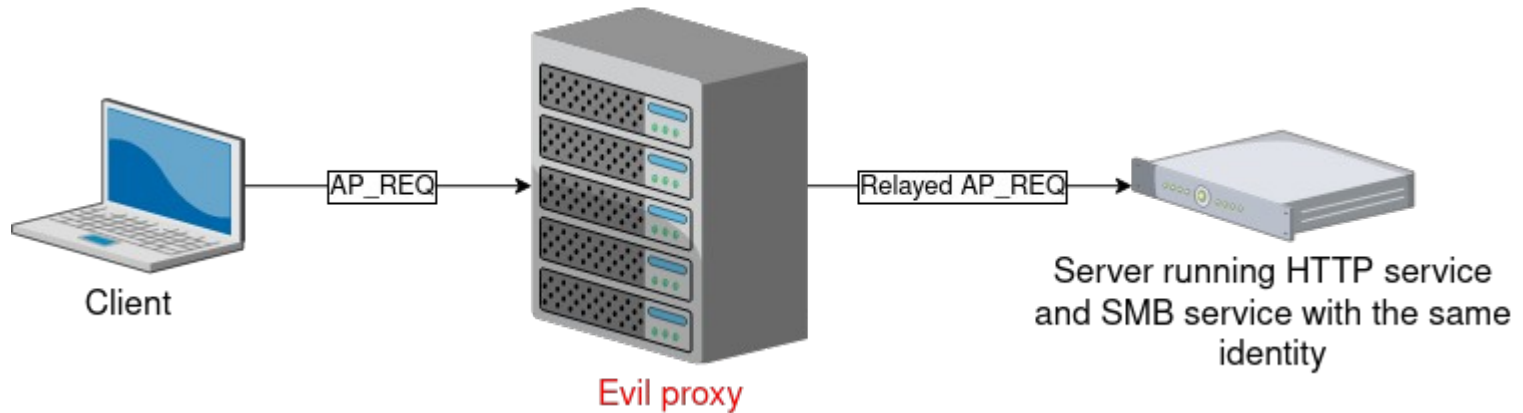
■ Security overview

- Two security measures to prevent replay attack
 - AP_REQ contain a timestamp : <5min
 - Host stores a MD5 hash of each AP_REQ : KRB_AP_ERR_REPEAT
- AP_REQ contains SPN of the service : **Not verified**



■ Security overview

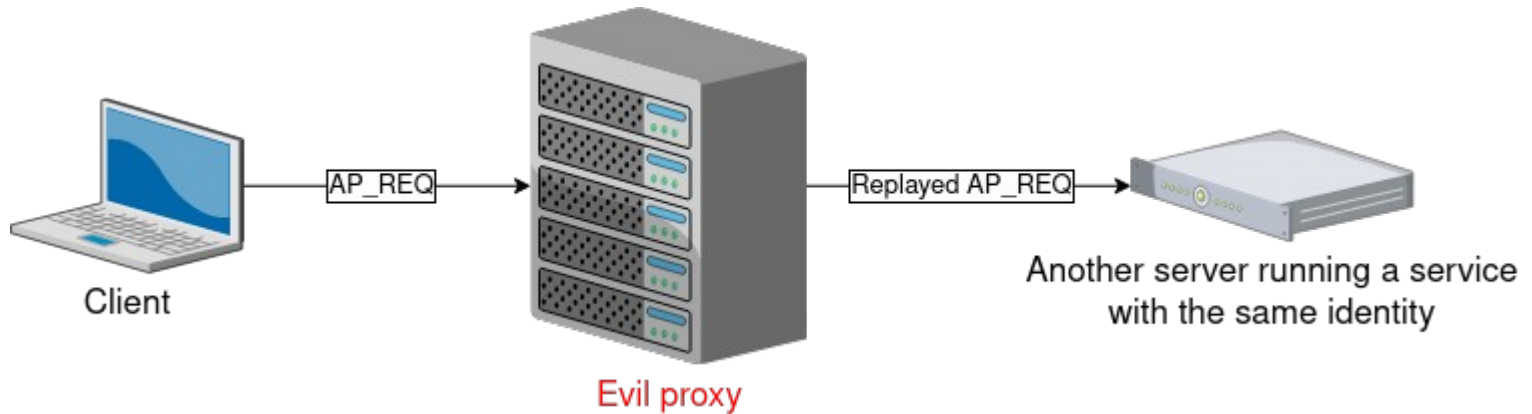
- Relay on a server using the same identity





■ Security overview

- Replay on another server using the same identity





■ Why do we need a proxy

- Still not supported by many clients (Firefox, ...)
 - No authentication possible if Kerberos is enforced
- How to use our tools against Kerberos protected websites?
 - BurpSuite
 - Certipy
 - ...

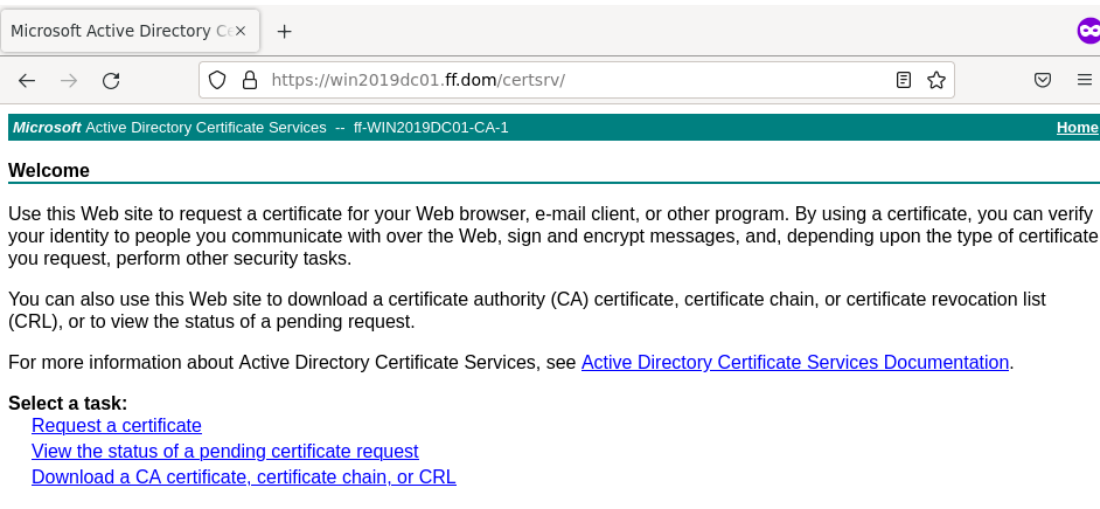
MitM Proxy – Prox-EZ (“prox easy”)








- **Prox-EZ implement Kerberos authentication**
 - Standard user/password capabilities
 - Pass-the-ticket capabilities (from TGT or ST)
 - Overpass-the-hash capabilities (from the NT hash)

MitM Proxy – Prox-EZ (“prox easy”)

Demo



Microsoft Active Directory C: x +

← → ↻   https://win2019dc01.ff.dom/certsrv/    ☰

Microsoft Active Directory Certificate Services -- ff-WIN2019DC01-CA-1 [Home](#)

Welcome

Use this Web site to request a certificate for your Web browser, e-mail client, or other program. By using a certificate, you can verify your identity to people you communicate with over the Web, sign and encrypt messages, and, depending upon the type of certificate you request, perform other security tasks.

You can also use this Web site to download a certificate authority (CA) certificate, certificate chain, or certificate revocation list (CRL), or to view the status of a pending request.

For more information about Active Directory Certificate Services, see [Active Directory Certificate Services Documentation](#).

Select a task:

- [Request a certificate](#)
- [View the status of a pending certificate request](#)
- [Download a CA certificate, certificate chain, or CRL](#)

```
[Prox-Ez Demo] $ getTGT.py 'ff.dom/brian:mysuperpassword1!'
Impacket v0.9.24 - Copyright 2021 SecureAuth Corporation

[*] Saving ticket in brian.ccache
[Prox-Ez Demo] $ export KRB5CCNAME=brian.ccache
[Prox-Ez Demo] $ klist
Ticket cache: FILE:brian.ccache
Default principal: brian@FF.DOM

Valid starting Expires Service principal
04/19/2023 16:27:42 04/20/2023 02:27:42 krbtgt/FF.DOM@FF.DOM
renew until 04/20/2023 16:27:42
[Prox-Ez Demo] $ python3 proxy.py -dc 'ff.dom/brian' -k -d
DEBUG:Proxy:Entered proxy, creating sockets.
DEBUG:Proxy:Proxy socket created.
DEBUG:Proxy:Binding proxy socket.
INFO:Proxy:Proxy socket bound, listening on 127.0.0.1:3128.
INFO:Proxy:Got connection from 127.0.0.1:43272.
DEBUG:Proxy:Creating new process.
DEBUG:Proxy:Handling connection.
DEBUG:Proxy.Client<->ProxyHelper:Creating new ClientToProxyHelper
DEBUG:Proxy.Client<->ProxyHelper:Our state: IDLE; their state: IDLE
DEBUG:Proxy.Client<->ProxyHelper:Received


CONNECT win2019dc01.ff.dom:443 HTTP/1.1
user-agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
proxy-connection: keep-alive
connection: keep-alive
host: win2019dc01.ff.dom:443
```

MitM Proxy – Prox-Ez (“prox easy”)



■ Available on GitHub:

- <https://github.com/synacktiv/Prox-Ez>
- PR & issues are welcome

☰ README.md 

Prox-Ez: The Swiss Army Knife of HTTP auth

This HTTP proxy handles all HTTP authentications on your behalf.

It supports NTLM EPA (channel binding and service binding), kerberos, pass-the-hash, overpass-the-hash (pass-the-key) and pass-the-ticket (TGT and TGS).



Any question?

Linked articles:

<https://www.synacktiv.com/publications/dissecting-ntlm-epa-with-love-building-a-mitm-proxy.html>

<https://www.synacktiv.com/publications/a-study-on-windows-http-authentication-part-ii.html>

<https://www.linkedin.com/company/synacktiv>

<https://twitter.com/synacktiv>

Our publications: <https://synacktiv.com>