SYNACKTIV

Dissecting NTLM EPA & Building a MitM proxy

PassTheSalt 2022

Whoami?

- **Pierre Milioni (*@b1two_*)**
  - Pentester at Synacktiv
- **Working for Synacktiv**
  - Offensive security
  - 100 ninjas: pentest, reverse engineering, development, CSIRT
  - 4 locations: Paris, Rennes, Lyon, Toulouse & remote
  - We are hiring! → *apply@synacktiv.com*

Introduction

- **A little bit of history**
  - Since 1993, introduced in 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

Agenda

- **NTLM & relaying**
- **NTLM-EPA**
- **MitM Proxy**
  - Why?
  - How?
  - Where?
  - Usage example

NTLM

- **NT 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 (new in 2022.6 – not tested)

- **… over HTTP**

Client's browser

Web server

Client initial HTTP request
GET / HTTP

Server requires authentication
401 + *WWW-Authenticate: NTLM*

Client NTLM Negotiate message
Authorization: NTLM <negotiate>

Server NTLM Challenge message
WWW-Authenticate: NTLM <challenge>

Client NTLM Auth message
Authorization: NTLM <auth>

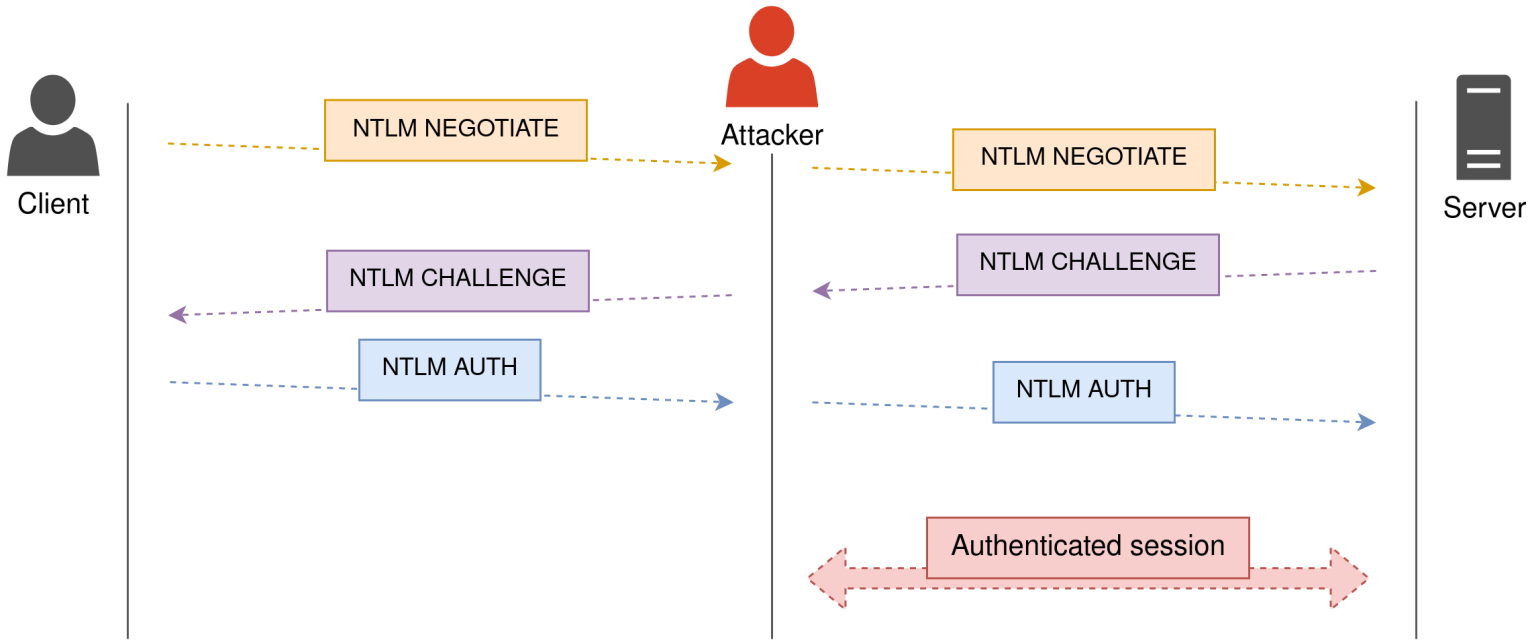
Server returns initially requested page
HTTP/1.1 200

More HTTP requests can be sent
in this authenticated TCP connection

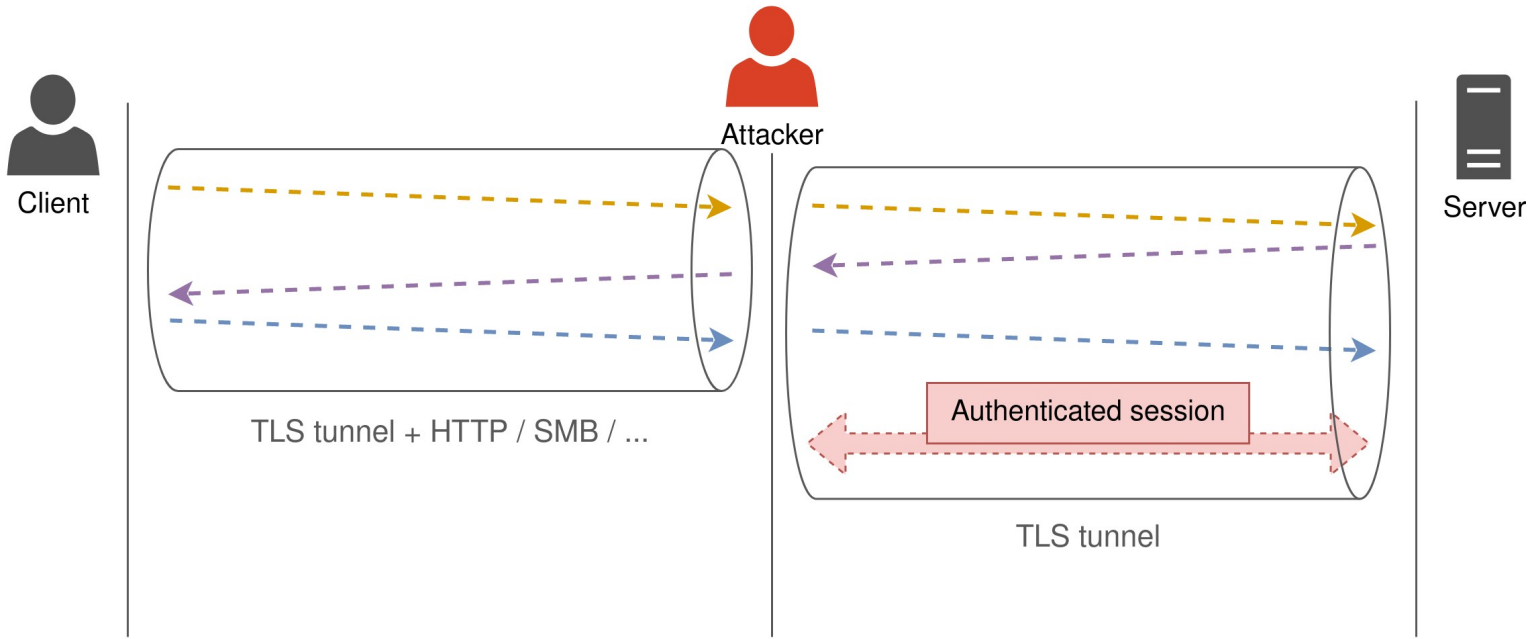
NTLM

NTLM relaying

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



- **NTLM relaying – HTTPS**
  - Attacker in a MitM position
  - Relays the client's authentication to the targeted server

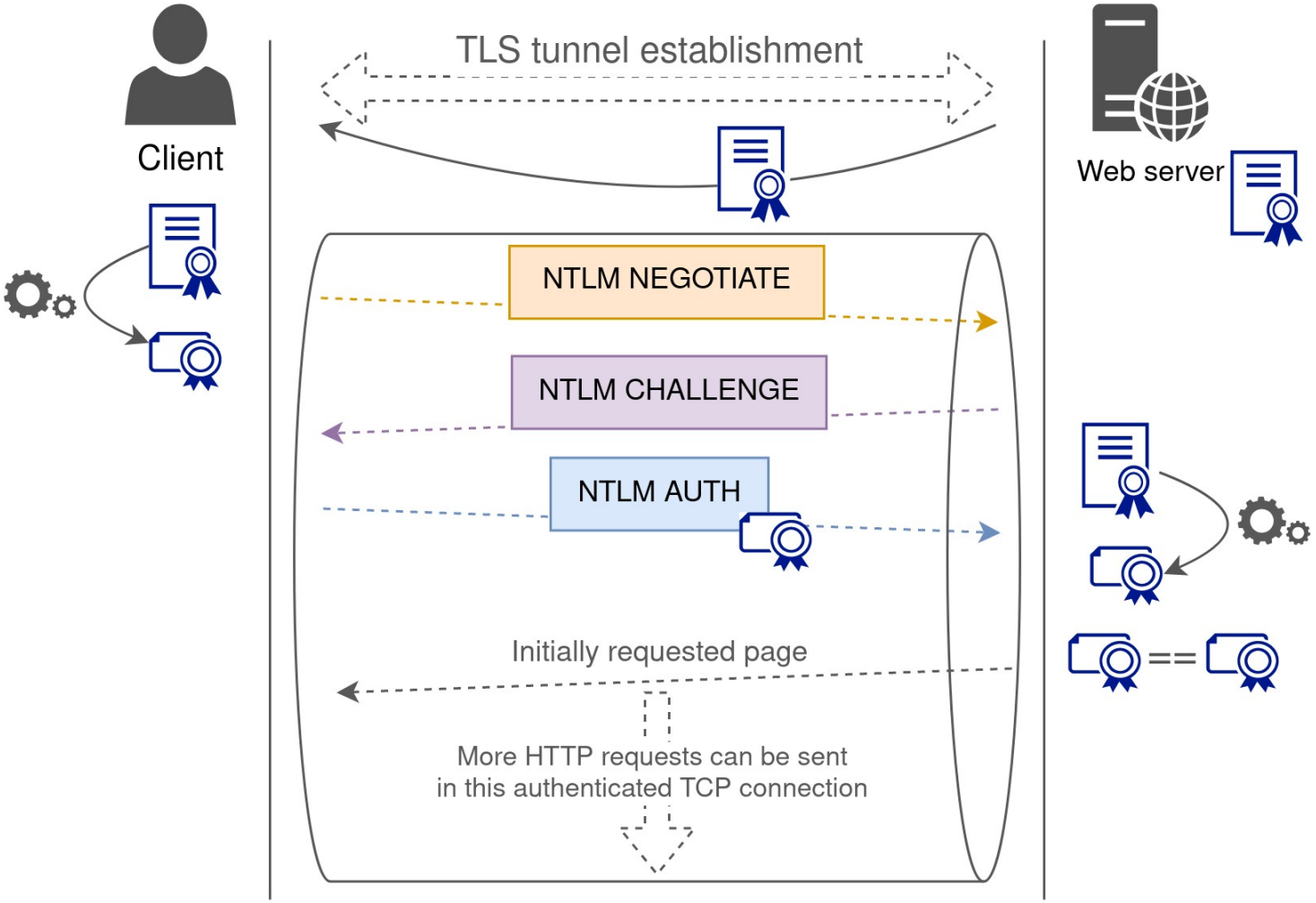


NTLM EPA

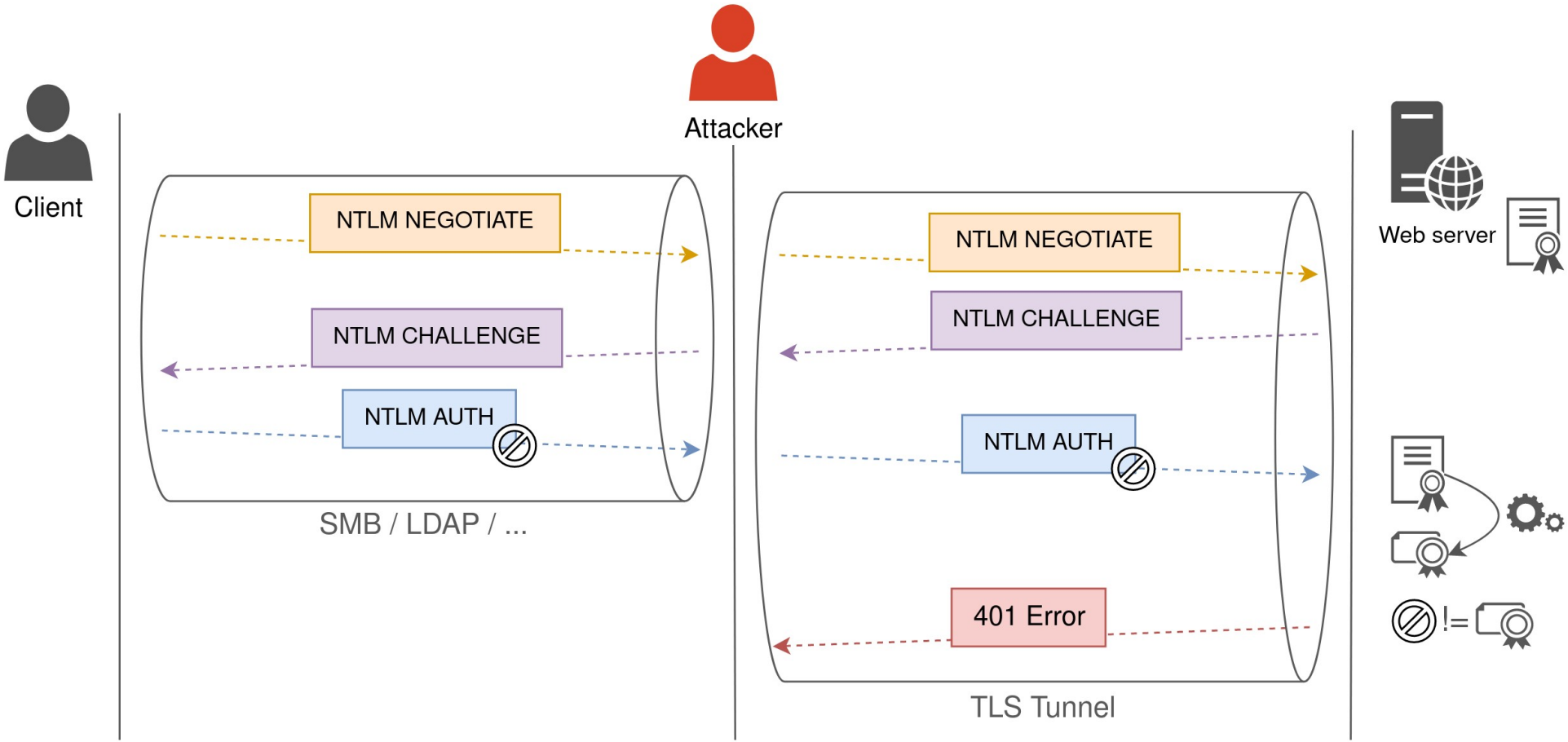
- **EPA – Extended Protection for Authentication**
  - Microsoft solution to protect against MitM attacks
  - Used on TLS based communications
  - “Binds” the authentication to the outer TLS channel

    → Adds a token that depends on the TLS tunnel into the NTLM authentication

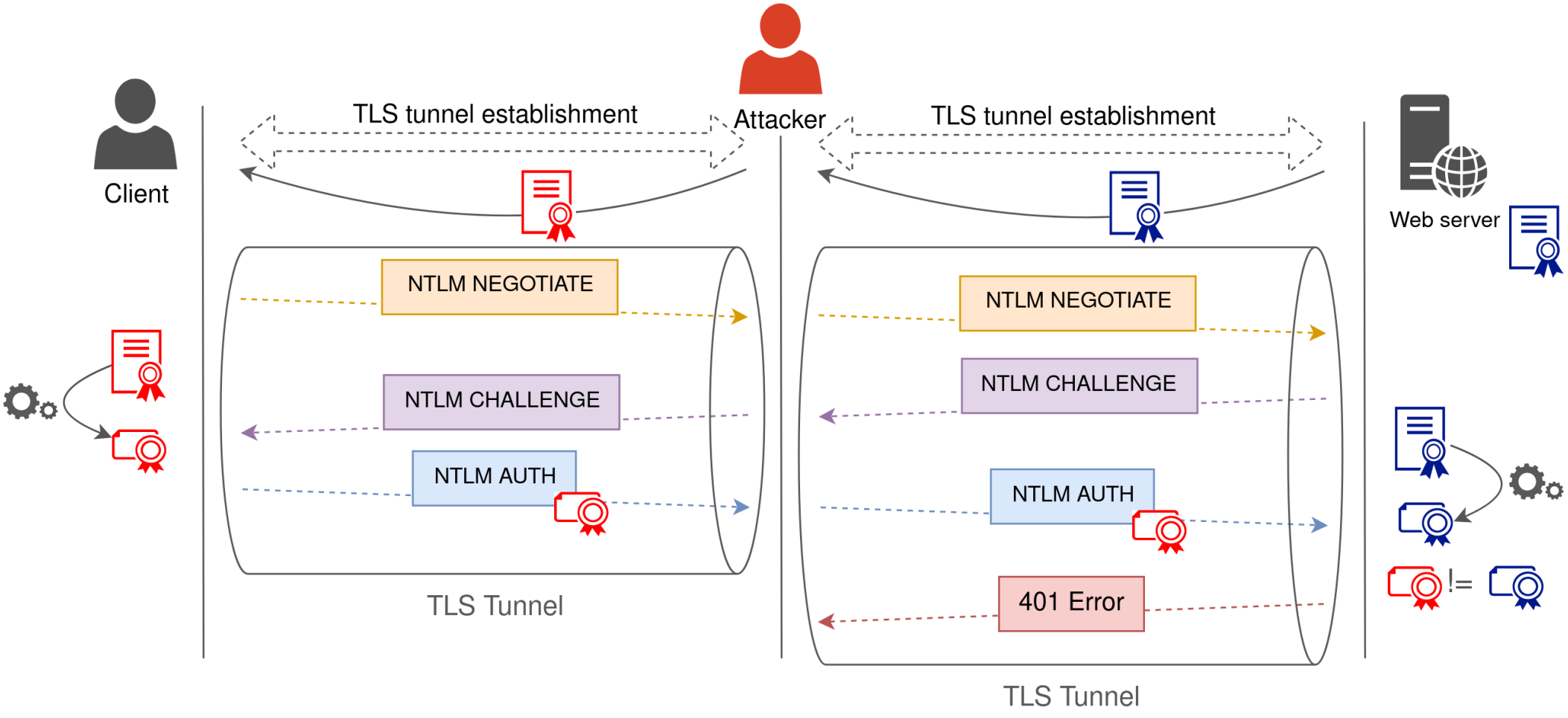
NTLM EPA


Client
TLS tunnel establishment
Web server
NTLM NEGOTIATE
NTLM CHALLENGE
NTLM AUTH
Initially requested page
More HTTP requests can be sent
in this authenticated TCP connection
==

NTLM EPA


Client
Attacker
Web server
NTLM NEGOTIATE
NTLM CHALLENGE
NTLM AUTH
SMB / LDAP / ...
NTLM NEGOTIATE
NTLM CHALLENGE
NTLM AUTH
401 Error
TLS Tunnel
!=

NTLM EPA


Client
Attacker
Web server
TLS tunnel establishment
TLS tunnel establishment
NTLM NEGOTIATE
NTLM CHALLENGE
NTLM AUTH
TLS Tunnel
NTLM NEGOTIATE
NTLM CHALLENGE
NTLM AUTH
401 Error
TLS Tunnel
!=

NTLM EPA

EPA – Extended Protection for Authentication

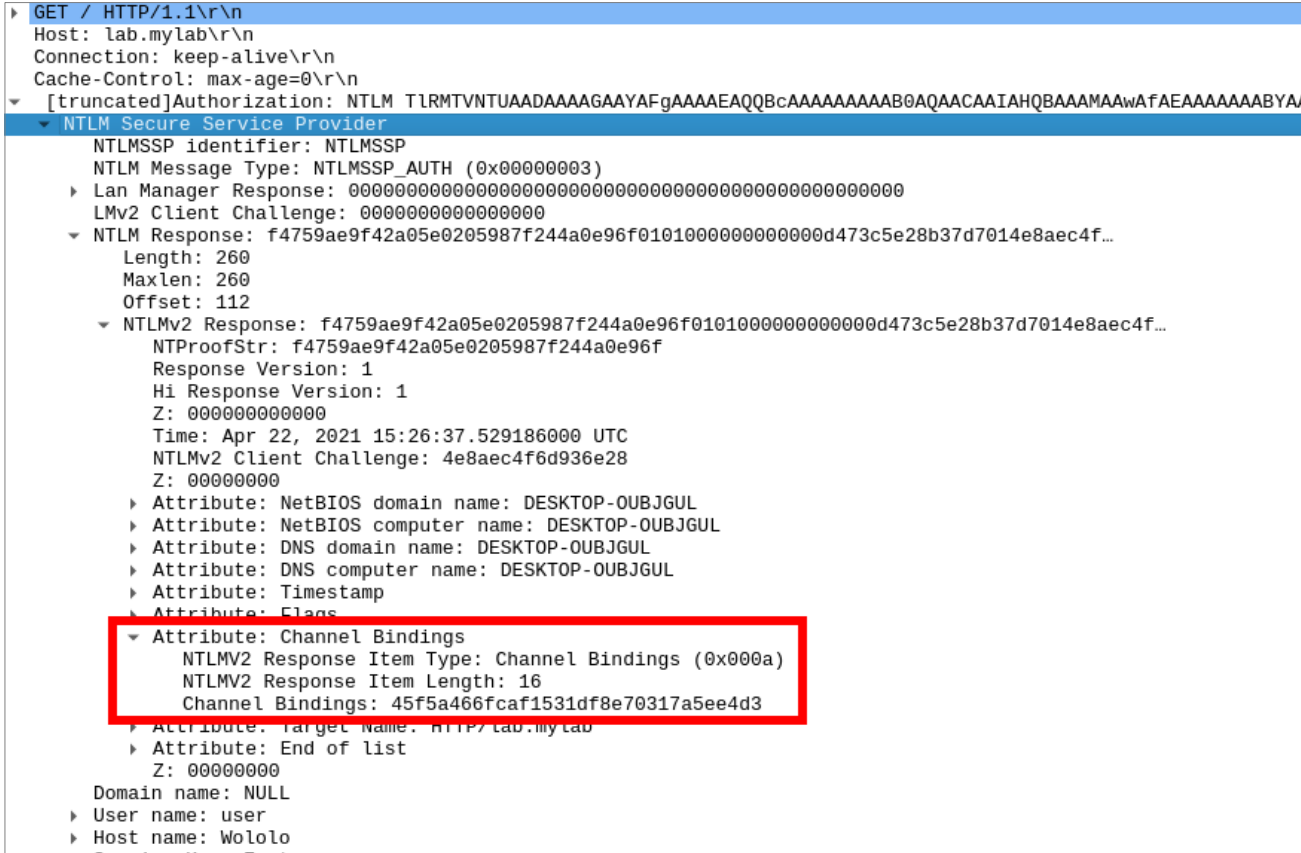
- CBT: *Channel Binding Token*
  - Hash of the server's certificate
  - With the hash function used to compute the certificate's signature

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 Bindings* attribute

- Derived from the CBT
- Inserted in the NTLM AUTH message



NTLM EPA

- **EPA – Extended Protection for Authentication**
  - 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")

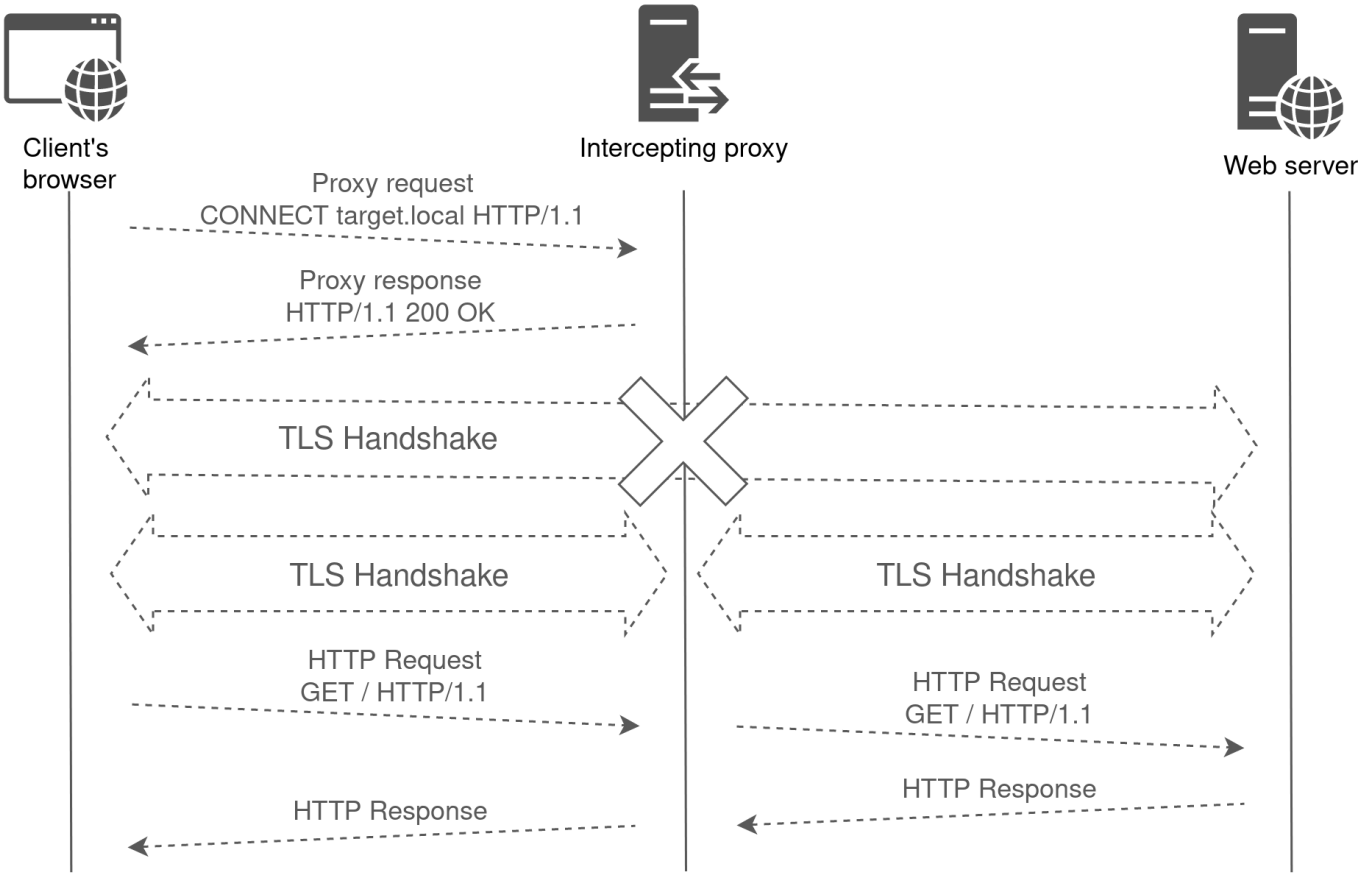
- **Why?**
  - Be able to use any tool against HTTPS servers using
    - NTLM
    - NTLM-EPA
    - *Kerberos*
  - Be able to control the authentication

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

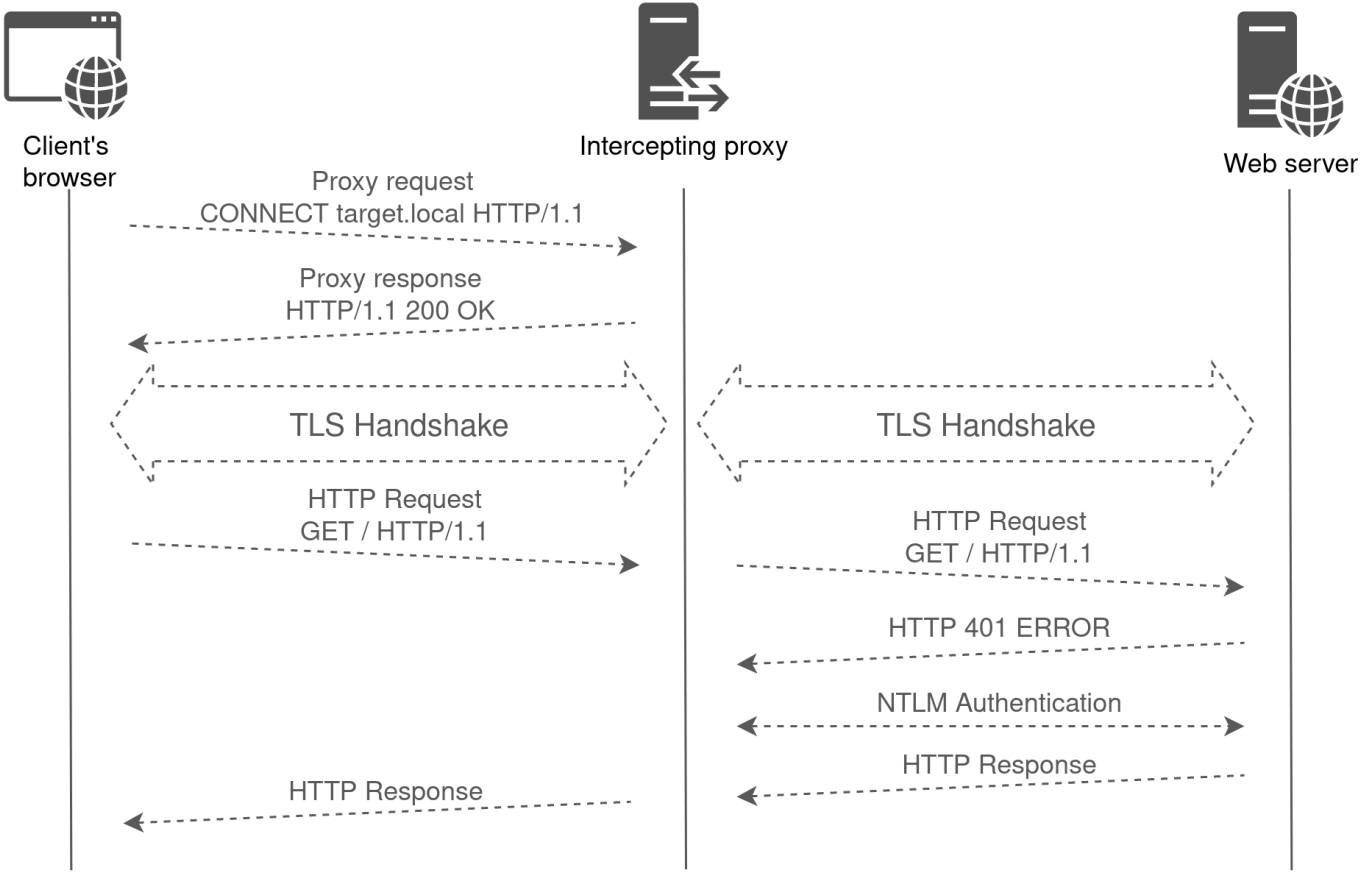
MitM Proxy – Prox-Ez ("prox easy")

- **How?**
  - TLS interception



MitM Proxy – Prox-Ez ("prox easy")

- **How?**
  - TLS interception



MitM Proxy – Prox-Ez ("prox easy")

- **Where?**
  - Available on GitHub:
    https://github.com/synacktiv/Prox-Ez
  - PR & issues are welcome

MitM Proxy – Prox-Ez ("prox easy")

Usage example

```
$ python3 proxy.py \
        --listen-address 127.0.0.1 \
        --listen-port 8088 \
        -du '.\user' \
        --hashes :45**************************21 \
        --debug
[…]

DEBUG:Proxy.Client<->ProxyHelper:Received
        CONNECT host:443 HTTP/1.1
        user-agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
        proxy-connection: keep-alive
        connection: keep-alive
        host: host:443
DEBUG:Proxy.Client<->ProxyHelper:Sending
        HTTP/1.1 200
```

MitM Proxy – Prox-Ez ("prox easy")

Usage example

[...]
DEBUG:Proxy.Client<->ProxyHelper:Received

GET /index.html HTTP/1.1
host: host
user-agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
accept-language: en-US,en;q=0.5
accept-encoding: gzip, deflate, br
connection: keep-alive
upgrade-insecure-requests: 1
sec-fetch-dest: document
sec-fetch-mode: navigate
sec-fetch-site: none
sec-fetch-user: ?1
if-modified-since: Sun, 03 Jul 2022 18:27:17 GMT
if-none-match: W/"fbfa286a8fd81:0"

MitM Proxy – Prox-Ez ("prox easy")

Usage example

```
[...]
DEBUG:Proxy.Proxy<->Server:Received

        HTTP/1.1 401 Unauthorized
        content-type: text/html
        server: Microsoft-IIS/10.0
        www-authenticate: NTLM
        date: Sun, 03 Jul 2022 18:43:54 GMT
        content-length: 58
```

MitM Proxy – Prox-Ez (“prox easy”)

Usage example

```
[...]
DEBUG:Proxy.Proxy<->Server:Sending

    GET /index.html HTTP/1.1
    host: host
    user-agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
    accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
    accept-language: en-US,en;q=0.5
    accept-encoding: gzip, deflate, br
    connection: keep-alive
    upgrade-insecure-requests: 1
    sec-fetch-dest: document
    sec-fetch-mode: navigate
    sec-fetch-site: none
    sec-fetch-user: ?1
    if-modified-since: Sun, 03 Jul 2022 18:27:17 GMT
    if-none-match: W/"fbfa286a8fd81:0"
    authorization: NTLM TlRMTVNTUAABAAAANYKI4AAAAAAAAAAAAAAAAAAAAAA=
```

MitM Proxy – Prox-Ez ("prox easy")

Usage example

```
[...]
DEBUG:Proxy.Proxy<->Server:Received

        HTTP/1.1 401 Unauthorized
        content-type: text/html; charset=us-ascii
        server: Microsoft-HTTPAPI/2.0
        www-authenticate: NTLM TlRMTVNTUAACAAAAC[...]7YAQAAAAA=
        date: Sun, 03 Jul 2022 18:43:54 GMT
        content-length: 341
```

MitM Proxy – Prox-Ez ("prox easy")

Usage example

```
[...]
DEBUG:Proxy.Proxy<->Server:Sending

    GET /index.html HTTP/1.1
    host: host
    user-agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
    accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
    accept-language: en-US,en;q=0.5
    accept-encoding: gzip, deflate, br
    connection: keep-alive
    upgrade-insecure-requests: 1
    sec-fetch-dest: document
    sec-fetch-mode: navigate
    sec-fetch-site: none
    sec-fetch-user: ?1
    if-modified-since: Sun, 03 Jul 2022 18:27:17 GMT
    if-none-match: W/"fbfa286a8fd81:0"
    authorization: NTLM TlRMTVNTUAADAAAAGAAYAEoAAACaAJoAYgAAAA0Ab[...]CKenSbwLbjzQTg==
```

MitM Proxy – Prox-Ez ("prox easy")

Usage example

```
[...]
DEBUG:Proxy.Proxy<->Server:Received

        HTTP/1.1 200 OK
        content-type: text/html
        last-modified: Sun, 03 Jul 2022 18:27:17 GMT
        accept-ranges: bytes
        etag: W/"fbfa286a8fd81:0"
        server: Microsoft-IIS/10.0
        persistent-auth: true
        date: Sun, 03 Jul 2022 18:43:54 GMT
        content-length: 44
```

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

https://www.linkedin.com/company/synacktiv
https://twitter.com/synacktiv
Our publications: https://synacktiv.com