TeleTrusT Information Security Professional

TeleTrusT Professional for Secure Software Engineering

T.I.S.P./T.P.S.S.E. Community Meeting 2025

Berlin, 04. - 05.11.2025

Das moderne SMB-Protokoll

Paul Blenderman, Secorvo Security Consulting GmbH

T.I.S.P. TeleTrusT Information Security Professional

T.P.S.S.E. TeleTrusT Professional for Secure Software Engineering

Microsoft Active Directory

Active Directory Certificate Services

35 Jahre System- und Netzwerkadministration

Seit Juli 2023 bei Secorvo

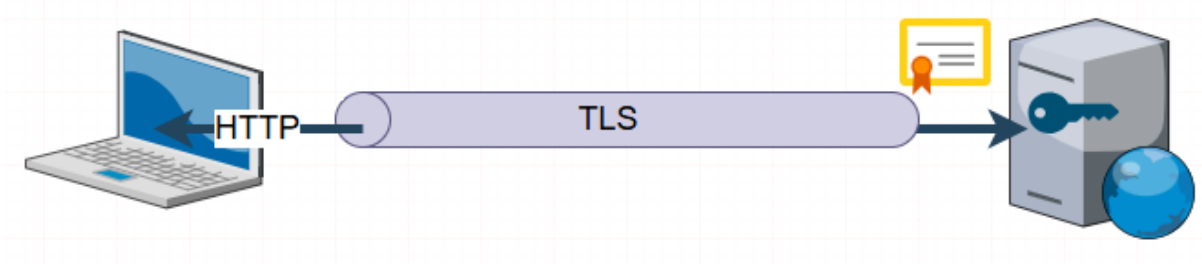
secorvo security consulting

Zum Vergleich: HTTP

V1.0 von 1992

Schon 1994 entwickelte Netscape „Secure Sockets Layer“, SSL 3.0 in 1996

Aktuell **TLS 1.2** (von **2008**) und **TLS 1.3** (von **2018**)



Verschlüsselt -> **Vertraulichkeit**

Serverauthentifizierung mittels Zertifikat,

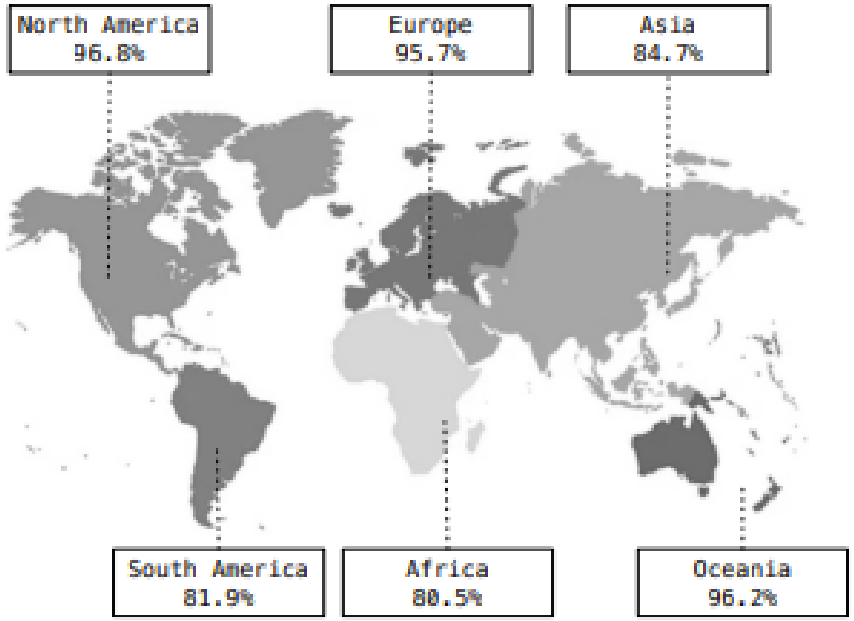
Signierung der Inhalte -> **Authentizität** + **Integrität**

Auch für SMTP, LDAP, IMAP, DNS, SIP, sogar FTP, POP3, NNTP…

Im Internet seit Jahren selbstverständlich.

(Let‘s Encrypt-Initiative in 2015 nach Snowden-Enthüllungen)

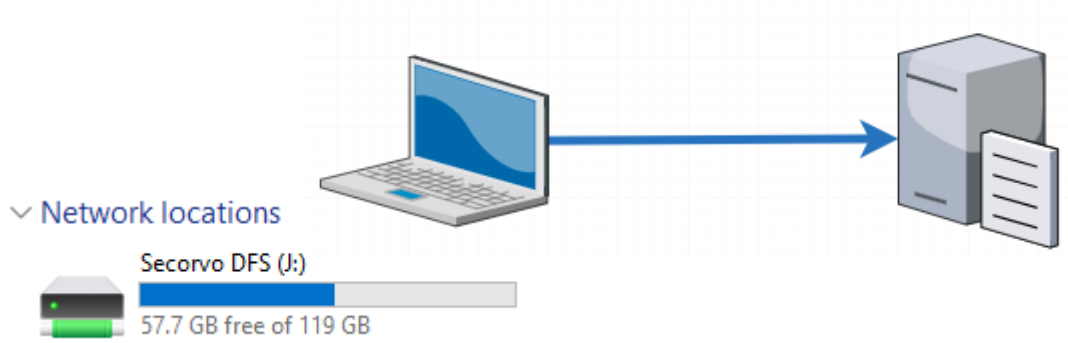
… und auch im internen Netz



% HTTPS [Mozilla, 2025]

SMB – Das Server Message Block-Protokoll

SMB ermöglicht den Zugriff auf Dateien über das Netzwerk.
„Netzwerklaufwerke“

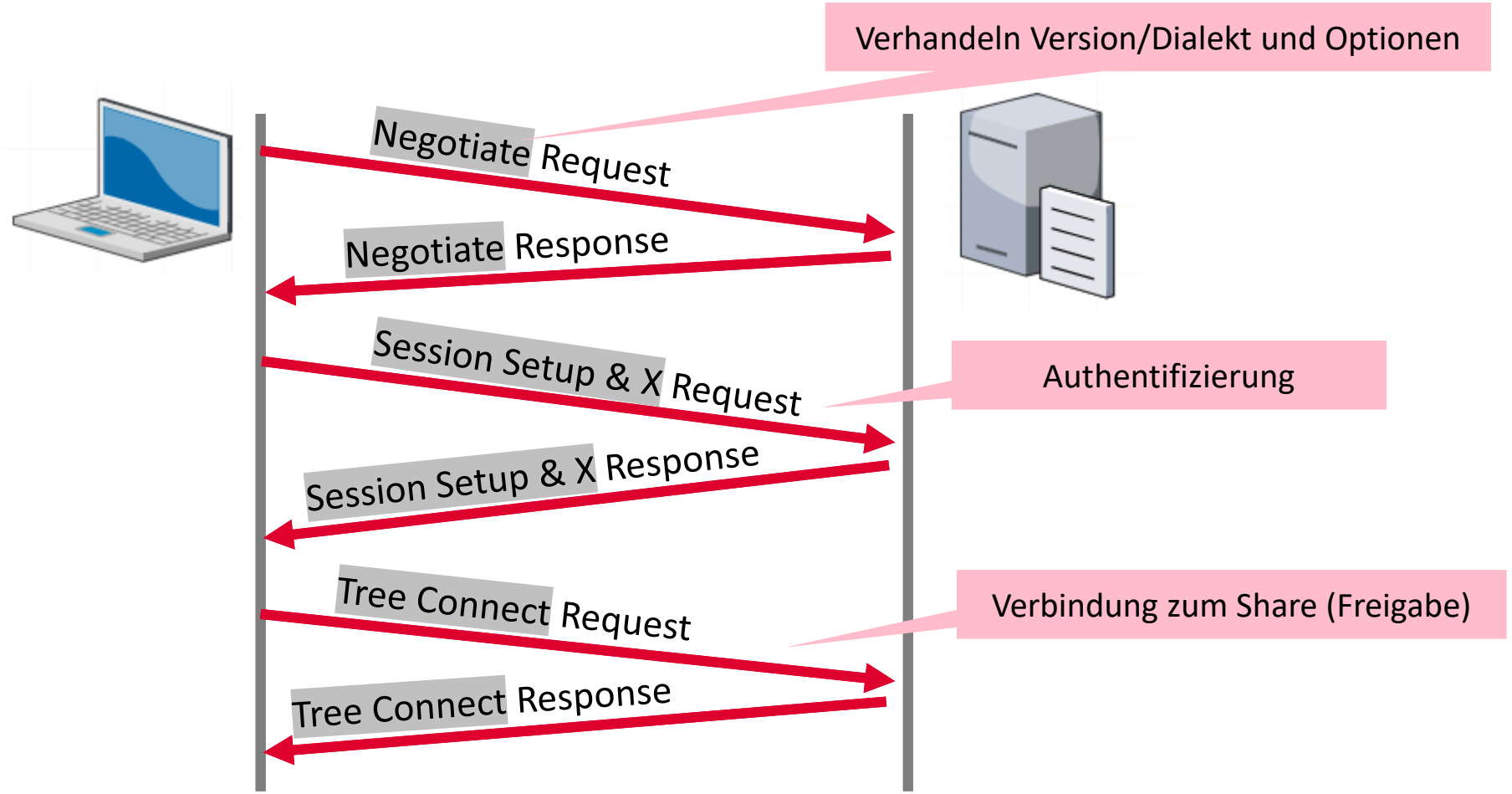


Open Close Read Write Delete Rename Create Directory Search Lock usw.

Außerdem Remote Procedure Calls und Druckaufträge.

Transport erfolgt über TCP und alternativ ab Windows 11/2022 über QUIC.

SMB - Verbindungsaufbau



Das historische SMB-Protokoll

Von IBM 1983 entwickelt für PC-DOS.
Noch kein TCP/IP (sondern NETBEUI).
Kleine LANs, kein Internet.

Ursprünglich in Assembler programmiert.

Sicherheit war damals kein großes Thema.

Version 1, auch SMB1 genannt, wurde aber in OS/2 und Windows NT übernommen, im Open Source-Projekt Samba implementiert, in NetApp Filers, usw.

SMB1

PCLAN1.0
Core, 1983, IBM

1.03
CorePlus

LANMAN1.0
OS/2, 1987

LANMAN1.2
OS/2, 1989

LANMAN2.0
OS/2, 1992

LANMAN2.1
OS/2, 1990

NT LM 0.12
1996, Windows NT 4.0

NetBT
(NetBIOS-over-TCP/IP)

Browser Service
LAN Manager 2, 1992, Network discovery

WINS
Windows NT 3.5, 1993, Name resolution

NetBIOS
Network API, 1983, IBM

LLMNR
Vista/Server 2008, Name resolution

UDP Port 137/138

TCP Port 139

TCP Port 445
Windows 2000

NetBEUI
Transport Protocol, 1983, IBM

Seit 2007 öffentliche Spezifikation, aktuell 54.0, 180 Seiten. Mehr als 100 (*Sub*-)*Commands*.

SMB2

Mit Vista bekommt Windows einen ganz neuen SMB-Client und -Server. Version 2 = **SMB2**.
Für die Rückwärts-Kompatibilität bleiben die SMB1-Komponenten aber enthalten.

```
PS> Get-ChildItem C:\windows\system32\drivers\srv*.sys, C:\windows\system32\drivers\mrx*.sys |
    Format-Table Fullname, @{Label = 'Description'; Expression = {$_.VersionInfo.FileDescription}}

FullName                                  Description
--------                                  -----------
C:\windows\system32\drivers\srv.sys       Server driver
C:\windows\system32\drivers\srv2.sys      Smb 2.0 Server driver
C:\windows\system32\drivers\srvnet.sys    Server Network driver
C:\windows\system32\drivers\mrxsmb.sys    Windows NT SMB Minirdr
C:\windows\system32\drivers\mrxsmb10.sys Longhorn SMB Downlevel SubRdr
C:\windows\system32\drivers\mrxsmb20.sys Longhorn SMB 2.0 Redirector
```

= SMB 1.0 Server

= SMB 1.0 Client

Longhorn = Vista

Nur noch 10 *Commands*. Aktuelle öffentliche Spezifikation 83.0 hat allerdings 497 Seiten.

Mit Windows 8 kommt dann Version ~~2.2~~ **3**. Erstmals gibt es *optional* auch Verschlüsselung.

SMB2 - Versionen

Version	Ab Windows	Wichtige Verbesserungen
1.0	-	-
2.0	Vista, 2008	Pipelining: bessere Performance, besonders im WAN Signing mit **HMAC-SHA-256** statt MD5
2.1	7, 2008 R2	Verbessertes Opportunistic Locking Support für große MTU
3.0	8, 2012	Transparent Failover (für Failover Cluster) **Encryption** mit AES-128 CCM Signing mit **AES-CMAC** **Secure Dialect Negotiation**
3.0.2	8.1, 2012 R2	Erweitertes Event-Logging, **abschaltbares SMB1**.
3.1.1	10, 2016	Encryption mit AES-128 **GCM** Signing mit AES-**GMAC** **Pre-Authentication Integrity** ersetzt Secure Dialect Negotiation
(3.1.1)	11, 2022	AES-256-GCM und -CCM-Encryption Pre-Authentication Integrity mit SHA-512

MD5 galt schon damals als gebrochen

Erstmals Verschlüsselung

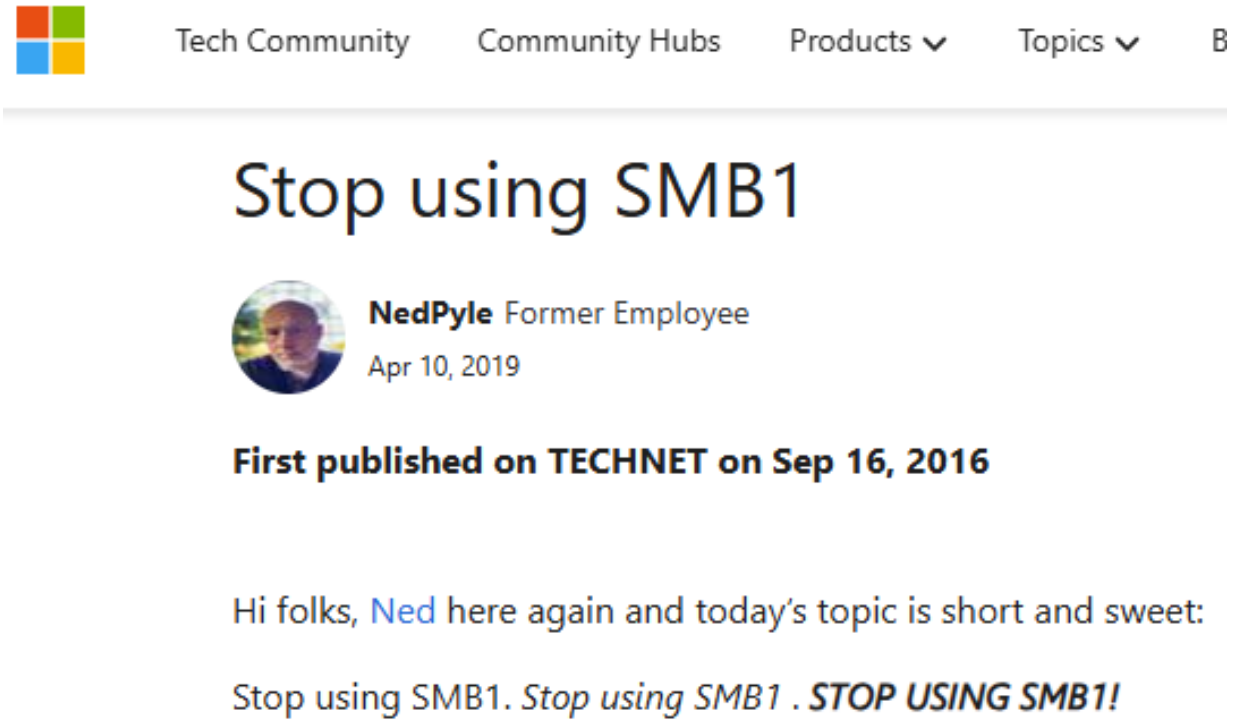
Schutz vor Downgrade-Angriffen

Bessere Performance

Vollständiger Schutz vor Downgrade-Angriffen

Stop using SMB1!

Ned Pyle, damals Program Manager für SMB bei Microsoft, schrieb 2016 den Appell, das hoffnungslos *kaputte* unsichere SMB1 abzuschaffen.



Das war ein Jahr vor WannaCry, das den aus dem NSA geleckten Exploit *EternalBlue* verwendete. Dieser nutzte eine Schwachstelle in SMB1 aus…

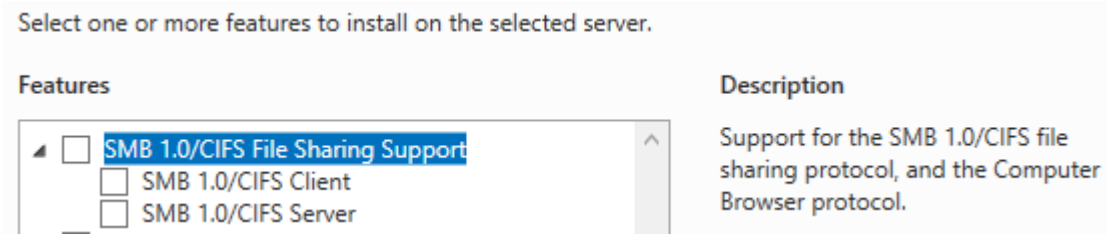
SMB1 abschalten

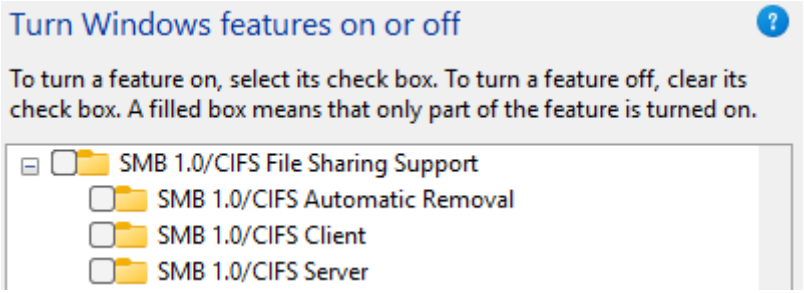
SMB1-Client und Server sollten auf Servern und Clients deaktiviert werden.

Nicht nur **abschalten**, sondern besser **deinstallieren**. Ab Windows 8 (2012):

```
PS> Disable-WindowsOptionalFeature -Online -FeatureName SMB1Protocol
```

Oder per Server Manager respektive Settings:





Seit Windows 10 1809 wird der SMB1-Client oder -Server automatisch entfernt, wenn er 15 Tage nicht genutzt wird.

Ab Windows 11 und Server 2022 ist SMB1 standardmäßig nicht installiert.

SMB1 abschalten

Der SMB1-**Server** lässt sich auch **abschalten** (sogar ab Windows 7/2008 R2):

```
PS> Set-ItemProperty –Path HKLM:\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters `
    -Name SMB1 -Type DWORD -Value 0
```

Oder ab Windows 8/2012:

```
PS> Set-SmbServerConfiguration -EnableSMB1Protocol $false
```

Es gibt keine entsprechende Group Policy. Aber natürlich geht eine Preference:

Name	Order	Action	Hive	Key	Value Name	Type	Value Data
SMB1	1	Update	HKEY_LOCAL_MACHINE	SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters	SMB1	REG_DWORD	00000000

Die Treiber lassen sich auch stilllegen (auch per Group Policy) – nicht unterstützt!

```
PS> Stop-Service srv, mrxsmb10
PS> Set-Service srv -StartupType Disabled
PS> Set-Service mrxsmb10 -StartupType Disabled
```

SMB Signing

Wie bei TLS: stellt **Integrität** und **Authentizität** sicher.

Eigentlich: symmetrischer Message Authentication Code (MAC)

Schlüssel (Session Key) wird aus Authentifizierungsverfahren abgeleitet. Nur so sicher wie die Authentifizierung. Kerberos verwenden!

Version	Ab Windows	MAC-Verfahren
1.0	-	MD5
2.0	Vista, 2008	**HMAC-SHA-256**
3.0	8, 2012	**AES**-CMAC
3.1.1	10, 2016	AES-**G**MAC
3.1.1	11, 2022	AES-**256**-GCM

Noch heute ungebrochen

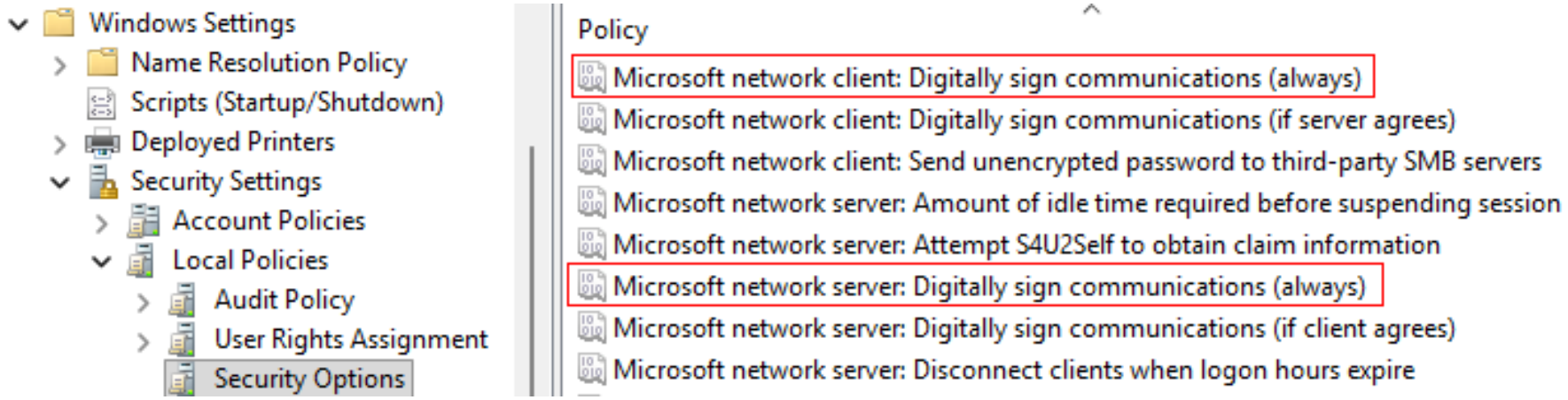
Effizienter

Effizienter, wenn CPU-Support vorhanden

Erst Windows 11 24H2/2022 erzwingen als Client standardmäßig SMB Signing.

SMB Signing erzwingen

Seit Windows 2000 per Group Policy



Ab Windows 8/2012 auch:

```
PS> Set-Smb‌Server‌Configuration –RequireSecuritySignature $true
PS> Set-Smb‌Client‌Configuration -RequireSecuritySignature $true
```

Setzt voraus, dass alle Gegenseiten Signing unterstützen.

SMB Encryption

Wie bei TLS: stellt **Vertraulichkeit** sicher.

Wie bei Signing: Schlüssel (Session Key) wird aus Authentifizierungsverfahren abgeleitet. Nur so sicher wie die Authentifizierung. Kerberos verwenden!

Version	Ab Windows	Verschlüsselungsverfahren
3.0	8, 2012	AES-128 CCM
3.1.1	10, 2016	AES-128 **G**CM
(3.1.1)	11, 2022	AES-**256**-GCM/-CCM

Effizienter, wenn CPU-Support vorhanden

Bis heute optional!

SMB Encryption anfordern und erzwingen

Ab Windows 11/Server 2025:

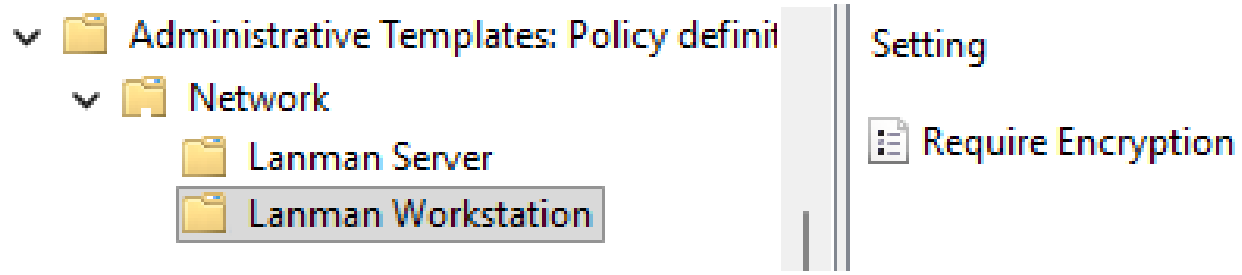
```
PS> Set-SmbClientConfiguration –RequireEncryption $true
PS> Set-SmbServerConfiguration –EncryptData $true
PS> Set-SmbServerConfiguration –RejectUnencryptedAccess $true
```

`$true` ist Standardwert.
`$false`: unverschlüsselte Verbindungen trotzdem akzeptieren, wenn `EncryptData` bei Server oder Share `$true` ist.

Nur für einen Share:

```
PS> Set-SmbShare -Name Finanzdaten -EncryptData $true
```

Oder per Group Policy (nur Client-Seite):



Setzt voraus, dass alle Gegenseiten Encryption unterstützen.

SMB Signing, Encryption und Performance

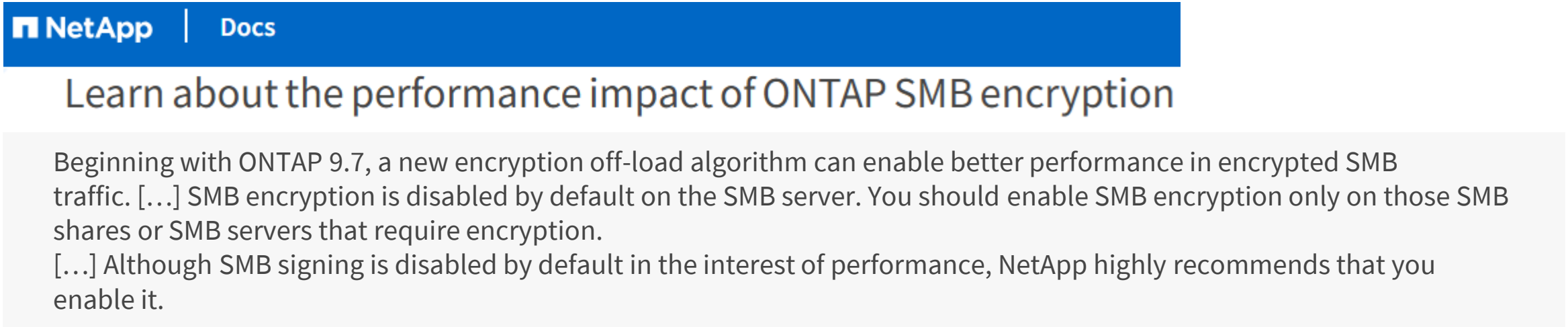
Wie bei TLS auch: Hasherzeugung und Ver-/Entschlüsselung sind rechenintensiv.

Intel und AMD-Prozessoren enthalten seit 2011 **A**dvanced **E**ncryption **S**tandard **N**ew **I**nstructions (**AES-NI**).

AES-**GCM** (**G**alois **C**ounter **M**ode) ist effizienter als **CCM** (**C**ipher **B**lock **C**haining).

SMB-Server und -Client 3.0 verwenden AES-NI.

Samba und viele Hersteller von Speichersystemen zogen erst spät nach:



SMB-Version erzwingen

Version	Ab Windows	Wichtige Verbesserungen
3.0	8, 2012	**Secure Dialect Negotiation**
3.1.1	10, 2016	Encryption mit AES-128 GCM Signing mit AES-GMAC **Pre-Authentication Integrity**

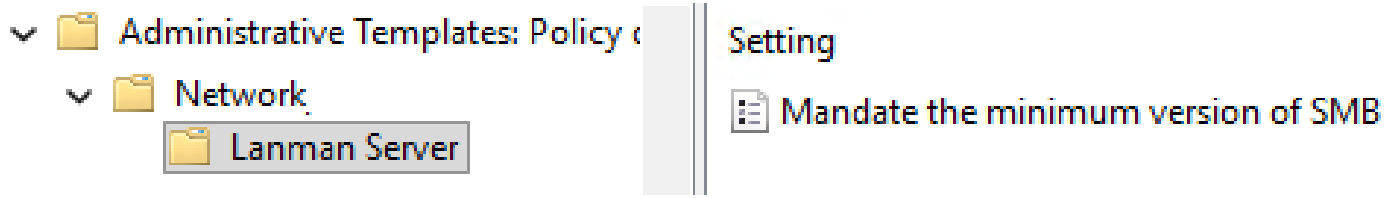
Schutz vor Downgrade-Angriffen

Vollständiger Schutz vor Downgrade-Angriffen

Ab Windows 11 24H2 und Server 2025:

```
PS> Set-Smb<b>Server</b>Configuration -Smb2DialectMin SMB311
PS> Set-Smb<b>Client</b>Configuration -Smb2DialectMin SMB311
```

Group Policy:



Bei Windows 10, Server 2019 und 2022, **nur für den Client**:

```
PS> New-ItemProperty –Path `
    HKLM:\SYSTEM\CurrentControlSet\Services\LanmanWorkstation\Parameters `
        -Name MinSMB2Dialect -PropertyType DWORD -Value 0x311
```

Inkompatible Gegenseiten aufspüren

Testen:

```
PS> New-SmbMapping -RemotePath \\server\share -RequireIntegrity $true -RequirePrivacy $true
```

Signing

Encryption

Prüfen auf Client-Seite:

```
PS> Get-SmbConnection | ft se*, sh*, dia*, si*, e*
ServerName                ShareName Dialect Signed Encrypted
----------                --------- ------- ------ ---------
srv25.testlab.secorvo.de Normal    3.1.1     True     False
srv.testlab.secorvo.de   Geheim    3.1.1    False      True
```

Encryption inkludiert Signing!

Prüfen auf Server-Seite:

```
PS> Get-SmbSession | ft cli*, di*
ClientComputerName ClientUserName   Dialect
------------------ --------------   -------
10.30.0.108        TESTLAB\user     3.1.1
```

Keine Angaben zu Signing und Encryption ☹

SMB Logging

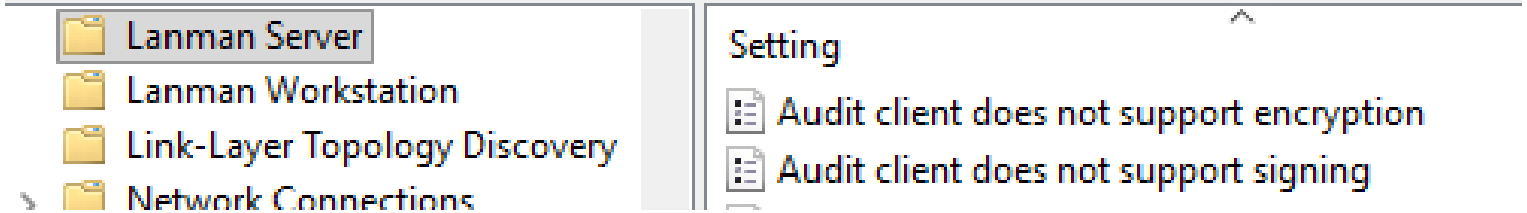
Seit Windows 10/Server 2016 kann der **Server** SMB1-Verbindungen melden:

```
PS> Set-SmbServerConfiguration -AuditSmb1Access $true
```

Erst mit Server 2025 kann das Scheitern des Aushandelns von Signing und/oder Encryption protokolliert werden:

```
PS> Set-SmbServerConfiguration -AuditClientDoesNotSupportEncryption $true
PS> Set-SmbServerConfiguration -AuditClientDoesNotSupportSigning $true
PS> Set-SmbClientConfiguration -AuditServerDoesNotSupportEncryption $true
PS> Set-SmbClientConfiguration -AuditServerDoesNotSupportSigning $true
```

Natürlich auch per Group Policy:



SMB Logging

Beispiel:

```
PS> Get-WinEvent -ProviderName Microsoft-Windows-SMBServer -max 1 | fl id, leveld*, mes*
Id                : 1003
LevelDisplayName  : Error
Message           : The server received an unencrypted message from client
                    when encryption was required. Message was rejected.
                    Client Address: 10.30.0.108:57997
```

Neu in Windows 2025:

```
PS> Get-WinEvent -ProviderName Microsoft-Windows-SMBServer -max 1 | fl id, leveld*, mes*
Id                : 1061
LevelDisplayName  : Warning
Message           : SMB2 Request Negotiate Dialect Failure
                    Client Address: 10.30.0.108:57997
                    Minimum dialect required by server: 0x311
                    Maximum dialect required by server: 0xFFFF
```

Leider erfahren wir nicht, was der Client wohl unterstützt…

NTLM abschaffen

Am besten grundsätzlich – ein ganz eigener Vortrag…

Ab Windows 11 24H2 und Server 2025 kann NTLM auch spezifisch beim SMB-Client abgeschaltet werden:

```
PS> Set-SmbClientConfiguration -BlockNTLM $true
```

Oder testen mit:

```
PS> New-SmbMapping -RemotePath \\server\share -BlockNTLM $true
```

Natürlich funktioniert Kerberos (noch!) nur innerhalb von (vertrauten) Forests.

- Keine Verbindungen zu IP-Adressen (z. B. \\10.20.30.40)!
- SMB-Eigenheit: keine CNAMEs (DNS-Alias) verwenden. Korrekt:

```
PS> netdom computername server /add:alias.domain.de
```

Das moderne SMB-Protokoll

- SMB1 deinstallieren
- SMB 3.1.1 erzwingen
- Signing erzwingen
- Encryption erzwingen
- NTLM abschaffen (sowieso!)

Dann kann man durchaus vom modernen SMB-Protokoll sprechen!

- Ausnahmen für Legacy-Systeme
- Vieles geht schrittweise
- Protokollierung nutzen