

## What's new in Domino 12.0.2 Security

OpenNTF, Feb 2023




Daniel Nashed -- <https://blog.nashcom.de>

# Agenda



- **Basic Agenda**

- Domino security
- e-mail security
- VSS Backup Writer support on Windows
- Demos & examples powered by **DNUG**  **LAB**

- **Disclaimer**

- Not a complete list of all new features!
- Detailed slides are available for reference

# Domino on Linux & Docker

- **Platform support**

- Support for **RHEL 9.x** and **SLES 15.3/15.4**
- Support for Linux **Kernel 5.x**
- Support for **SELinux** in **enforced mode**

- **Domino on Docker**

- New Container image based on the HCL Community image including Nash!Com Domino Start Script
- <https://github.com/hCL-TECH-SOFTWARE/domino-container>
- Special build on RedHat Universal Base Image 8.6 (Traveler and Domino Leap)
- The community project offers still many more options
  - Including **Nomad Server, Verse, REST API** install option

# Important Software Package Updates

- **OpenSSL 3.0.5**

- New major OpenSSL version
- Modular design helps with **FIPS 140-2** support
  - <https://www.openssl.org/blog/blog/2022/08/24/FIPS-validation-certificate-issued/>
- Starting with Notes/Domino 12.0.2 OpenSSL is linked into core with no separate .dll/.so files!

- **LibCurl 7.83.0**

- Important package, leveraging OpenSSL
- Linked into core Notes/Domino since 10.x
- Used from Lotus Script and also in the back-end for other features (CertMgr, OIDC)

- **Apache Tika 2.4.1**

- Used for attachment filtering when full text indexing attachments

- **Packages are newer than in most Linux distributions!**

# HCLSoftware

Domino Directory  
Trusted Roots Update



# New Trusted Roots

- Imported from LibCurl /local/notesdata/cacert.pem
- Additional information added to new Certifier documents for Internet Trusted Roots

**DOMINO DIRECTORY - Certifier [INTERNET CERTIFIER] :** GlobalSign/GlobalSign ECC Root CA - R5/GlobalSign

Basics | Recovery Configuration | Other | Administration

**Basics**

|                                   |   |
|-----------------------------------|---|
| Certifier type:                   | Internet Certifier  |
| Certifier name:                   | GlobalSign/GlobalSign ECC Root CA - R5/GlobalSign                               |
| Issued by:                        | GlobalSign/GlobalSign ECC Root CA - R5/GlobalSign                               |
| Issued to:                        | CN=GlobalSign/O=GlobalSign/OU=GlobalSign ECC Root CA - R5                       |
| Primary key identifier:           |   |
| International key identifier:     |   |
| Current key strength:             |   |
| Current key creation date:        |   |
| Subject Key Identifier (SHA1):    | 3DE6 2948 9BEA 07CA 2144 4A26 DE6E DED2 83D0 9F59                               |
| Certificate Fingerprint (SHA256): | 179F BC14 8A3D D00F D24E A134 58CC 43BF A7F5 9C81 82D7 83A5 13F6 EBEC 100C 8924 |
| Key Algorithm:                    | EC Public Key   |
| Key size:                         | 384   |
| Curve name:                       | secp384r1 (NIST P-384)  |
| Key Usage:                        | Key_CertSign CRL_Sign   |
| Signing Algorithm:                | ecdsa-with-SHA384   |
| Not Before:                       | 13.11.2012 01:00:00   |
| Not After:                        | 19.01.2038 04:14:07   |
| Certified public key:             |   |

# Improved Import/Export Dialogs

- Requires **Notes 12.0.2 Client** and fully supports ECDSA certificates

The screenshot displays the 'Import Internet Certificates' dialog box. The left pane shows a list of certificates with 'DigiCert Assured ID Root G3' selected. The right pane shows the 'Certificate information' for this certificate, with a red box highlighting the 'Activated' and 'Expires' fields.

**Import Internet Certificates**

Do you want to accept all certificates from the import file into the directory?

All Internet Certificates

| Type                        | Issued To                   | Issued By                   |
|-----------------------------|-----------------------------|-----------------------------|
| DigiCert Assured ID Root G3 | DigiCert Assured ID Root G3 | DigiCert Assured ID Root G3 |

Selected item:

Issued to: DigiCert Assured ID Root G3 (Email)

Issued by: DigiCert Assured ID Root G3 (Email)

Activated: 01/08/2013 Type: Internet certificate authority

Expires: 15/01/2038 Fingerprint: 7C7F 6531 0C81 DF8D BA3E 99E2 5

Advanced Details...

Accept All

This certificate belongs to an internet certificate authority.

Certificate information: select attribute to display details below

| Attribute              | Value  |
|------------------------|--|
| Issued to              | CN=DigiCert Assured ID Root G3/OU=www.digicert.com/O=DigiCert Inc/C=US       |
| Issued by              | CN=DigiCert Assured ID Root G3/OU=www.digicert.com/O=DigiCert Inc/C=US       |
| MD5 Fingerprint        | 7C7F 6531 0C81 DF8D BA3E 99E2 5CAD 6EFB                                      |
| SHA1 Fingerprint       | F517 A24F 9A48 C6C9 F8A2 0026 9FDC 0F48 2CAB 3089                            |
| SHA1 Key Identifier    | CBD0 BDA9 E198 0551 A14D 37A2 8379 CE8D 1D2A E484                            |
| Serial number          | 0BA1 5AFA 1DDF A0B5 4944 AFCD 24A0 6CEC                                      |
| Version                | Version 3  |
| Activated              | 01/08/2013   |
| Expires                | 15/01/2038   |
| Signature algorithm    | ecdsa-with-SHA384  |
| Key algorithm          | EC Public Key  |
| Key strength           | 384 bits   |
| Subject key identifier | 0414 CBD0 BDA9 E198 0551 A14D 37A2 8379 CE8D 1D2A E484                       |
| Key usage              | Digital signature, Certificate signing, CRL signing                          |
| Basic constraints      | Certificate issued to a certificate authority=TRUE, Path length constraint=0 |

CN= DigiCert Assured ID Root G3  
OU= www.digicert.com  
O= DigiCert Inc  
C= US

Close

# HCLSoftware

## CertMgr & CertStore

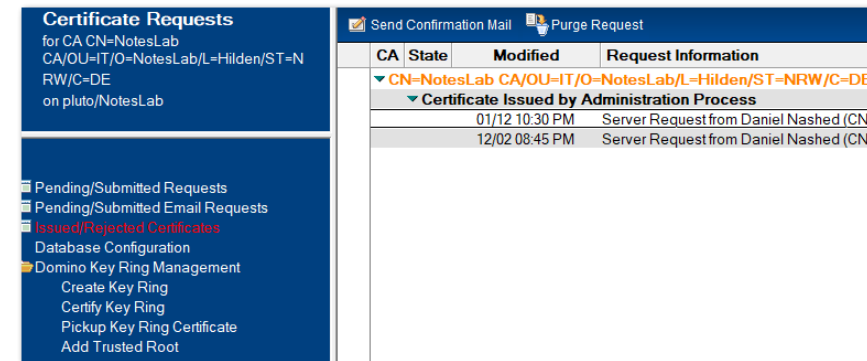
Domain wide trusted root,  
private key & certificate management





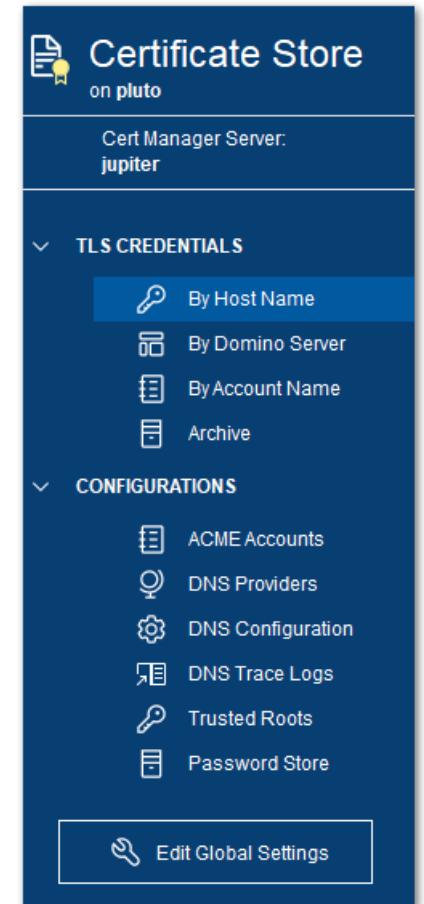
# Before Domino 12: kyr files, kyrtool & OpenSSL

- Domino used \*.**kyr** file format for Internet Certificates
  - Old IBM format nobody else can read or write
  - The only tool available to read and write is “**kyrtool**”
    - Very flexible but command line driven -- Not always easy to handle
    - Replaced old **certreq.nsf** database which wasn't easy to use either
- **Creating keys** and **CSR** required an external tool like “**openssl**” on **command line**
  - Very powerful, but also very cryptic tool with confusing command line for most admins
- \*.**kyr** files have a corresponding \*.**sth** containing the encoded password
  - Can be decoded with simple perl script
- Old **kyrfile cache** for internet processes always needed restart for any \*.**kyr** change



# certstore.nsf

- Domain wide database managed by **CertMgr** task
- **Secure**, automated deployment for TLS Credentials and trusted roots
- Private keys are **encrypted** with CertMgr server and the server specified in field “**Servers with access:**”
  - Special designed Vault style encryption with new API
- Easy to use with modern interface
- **CertMgr servertask** is only supported on **W64** and **Linux64**
  - **AIX** and **OS400** can still leverage **certstore.nsf** and the new TLS Cache
    - Create replica manually
  - **New in Domino 12.0.2: Full support for AIX**



# Create Domain wide certstore.nsf

- First server in domain starting the “**certmgr**” server task is setup as the CertMgr Server
  - Checks the Domino **directory profile** on **admin server** for an existing CertMgr server
  - If no server exists automatically creates the domain wide **certstore.nsf** database
  - Updates the directory profile on admin server to propagate the CertMgr server in the domain
- Starting the certmgr server task on any **additional** server in the domain creates a replica
  - Each additional server acts like a “**CertMgr client**” and will just replicate the database every 2 minutes
  - Keeping the CertMgr server task loaded is an optional convenience step
  - Any type of replication setup which ensures a short replication cycle can be used as well

# certstore.nsf – TLS Credentials

- **TLS Credential** = private key + leaf certificate + chain (intermediates) + trusted root
- Replaces “\*.kyr files”
  - Stored in **PEM** format (text with base64 encoded data)
- Can be created via
  - **ACME V2** protocol (Let's Encrypt & others)
  - Manual flows including import
  - Domino MicroCA (exportable in 12.0.2)
- Specify trusted roots used for client certificate verification
  - Used to be hidden in **kyr-file** and was difficult to manage

The screenshot shows the 'TLS Credentials' configuration page. At the top, there are tabs: 'Main', 'Security/Keys', 'Manual', and 'Comments'. The 'Main' tab is selected. Below the tabs, there is a table with the following fields and values:

|                         |   |
|-------------------------|---|
| Status:                 | Issued                                  |
| Host names:             | pluto.csi-domino.com                    |
| Servers with access:    | pluto/NotesLab <input type="checkbox"/> |
| Status:                 | Valid                                   |
| Certificate expiration: | Sun 05/30/2021 02:43:18 PM              |
| Certificate renew date: | Fri 04/30/2021 02:43:18 PM              |
| Certificate provider:   | ACME                                    |
| ACME account:           | LetsEncryptProduction                   |
| Key type:               | ECDSA                                   |
| Curve name:             | NIST P-384                              |
| Automatically renew:    | 30 days before expiration               |
| Request key rollover:   |   |
| Keyring file:           |   |

The screenshot shows the 'Trusted Roots' configuration page. On the left, there is a list of trusted roots with a search icon and the text 'CN=ISRG Root X1'. On the right, there is a 'Select Keywords' dialog box with a list of keywords and checkboxes:

| Select Keywords                     |   |
|-------------------------------------|---|
| Keywords                            |   |
| <input type="checkbox"/>            | CN=Fake LE Root X1  |
| <input checked="" type="checkbox"/> | CN=ISRG Root X1/O=Internet Security Research Group/C=US     |
| <input type="checkbox"/>            | CN=AAA Certificate Services/O=Comodo CA Limited/L=Salford/S |
| <input type="checkbox"/>            | CN=Buypass Class 2 Root CA/O=Buypass AS-983163327/C=NO      |
| <input type="checkbox"/>            | CN=ISRG Root X2/O=Internet Security Research Group/C=US     |

# Manual Certificate Operations

- **1. CertMgr** processes submitted requests and creates
  - Private key ( RSA or ECDSA)
    - Saved locally encrypted for assigned servers
- CSR (**C**ertificate **S**igning **R**quest) signed by private key→ PEM
- **2.** Admin copies CSR to CA
- **3.** Admin imports certificate & chain ( PEM ) back
- Paste full chain in any order and submits the form again
- Duplicate certs are ignored
- Missing intermediate certs and root are automatically added from “Trusted Roots” in **certstore.nsf**

The image displays three overlapping screenshots of the 'TLS Credentials' form in the CertMgr application. The top screenshot shows the 'Main' tab with fields for 'Host names' (www.notes.lab), 'Servers with access' (notes-lab-01/Srv/NotesLab), 'Certificate provider' (Manual), 'Key type' (ECDSA), 'Curve name' (NIST P-384), and 'Automatically renew' (30 days before expiration). The middle screenshot shows the 'Copy CSR' button highlighted in the top bar. The bottom screenshot shows the 'Manual' tab with a 'Paste - Certificates & Roots (PEM)' field and a 'Copy - CSR (Certificate Signing Request)' field containing a PEM-formatted CSR.

**Top Screenshot: TLS Credentials Form**

Buttons: Submit Request, Save & Close, Cancel

Tabs: Main | Security/Keys | Manual | Comments

Main

Status: [Dropdown]

Host names: [Text: www.notes.lab]

Servers with access: [Text: notes-lab-01/Srv/NotesLab]

Status: [Text]

Certificate expiration: [Text]

Certificate renew date: [Text]

Certificate provider: [Text: Manual]

Key type: [Text: ECDSA]

Curve name: [Text: NIST P-384]

Automatically renew: [Text: 30 days before expiration]

Keyring file: [Text]

**Middle Screenshot: TLS Credentials Form**

Buttons: Submit Request, Copy CSR, Paste Certificate, Edit

Tabs: Main | Security/Keys | Manual | Comments

Main

Status: [Text: Waiting]

Host names: [Text: www.notes.lab]

Servers with access: [Text: notes-lab-01/Srv/NotesLab]

**Bottom Screenshot: Manual Tab**

Tabs: Main | Security/Keys | Manual | Comments

Paste - Certificates & Roots (PEM)

Copy - CSR (Certificate Signing Request)

```
-----BEGIN CERTIFICATE REQUEST-----
MIIBIjCCARACAQAwVjELMAkGA1UEBhMCVVMxHzAdBgNVBAgMAk1B
DAZCb3N0b24xETAPBgNVBAoMCESvdGVzTGFiMRywFAYDVQQDDA13
bGFfMHYwEAYHKoZIzj0CAQYFK4EEACIDYgAEpSQ0qM/Z8q22Yycq
```

# certstore.nsf – Trusted Roots

- Stored in trusted, secured certstore.nsf
  - Replicated domain wide
- Used for client cert verification
- And auto complete certificate chains
  - ACME and manual flows
- Certificate chains are automatically sorted & completed
  - **Private Key** → matching **leaf certificate**  
→ **intermediate certs** in the right order → **trusted root**
- Tip: you can import intermediate certificates as “Trusted Root” to be used to auto complete chains

The screenshot displays the IBM Domino certstore.nsf interface. The top bar includes buttons for 'Add Trusted Root', 'Edit Document', and 'Delete Trusted Root'. Below this is a table listing trusted roots with columns for Name, Certificate Expiration, Type, and Curve/Size. The table is organized into sections: 'Imported' (containing one entry), 'ICAP' (containing one entry), and 'No usage category' (containing five entries). A modal window titled 'Trusted Root' is open, showing details for a specific certificate. The modal has tabs for 'Main', 'Certificates', and 'Comments'. The 'Main' tab is active, displaying the certificate's status as 'Issued', its name as 'CN=ISRG Root X2/O=Internet Security Research Group/C=US', and its usage categories. Below this, the 'Certificate Information' section provides details such as expiration date, activation date, algorithm, curve name, certificate fingerprint, signing algorithm, and key usage.

| Name  | Certificate Expiration | Type  | Curve/Size             |
|---|------------------------|-------|------------------------|
| <b>1 Imported</b>                                       |                        |       |                        |
| ⚠ CN=AAA Certificate Services/O=Comodo CA Limited/L=S   | 01.01.2029 00:59:59    | RSA   | 2048                   |
| <b>1 ICAP</b>   |                        |       |                        |
| 🔒 Fake LE Root X1/                                      | 23.03.2036 23:53:46    | RSA   | 4096                   |
| <b>5 -- No usage category --</b>                        |                        |       |                        |
| 🔒 Bypass Class 2 Root CA/Bypass AS-983163327/NO         | 26.10.2040 10:38:03    | RSA   | 4096                   |
| 🔒 DST Root CA X3/Digital Signature Trust Co.            | 30.09.2021 16:01:15    | RSA   | 2048                   |
| 🔒 ISRG Root X1/Internet Security Research Group/US      | 04.06.2035 13:04:38    | RSA   | 4096                   |
| 🔒 ISRG Root X2/Internet Security Research Group/US      | 17.09.2040 18:00:00    | ECDSA | secp384r1 (NIST P-384) |
| 🔒 (STAGING) Pretend Pear X1/(STAGING) Internet Security | 04.06.2035 13:04:38    | RSA   | 4096                   |

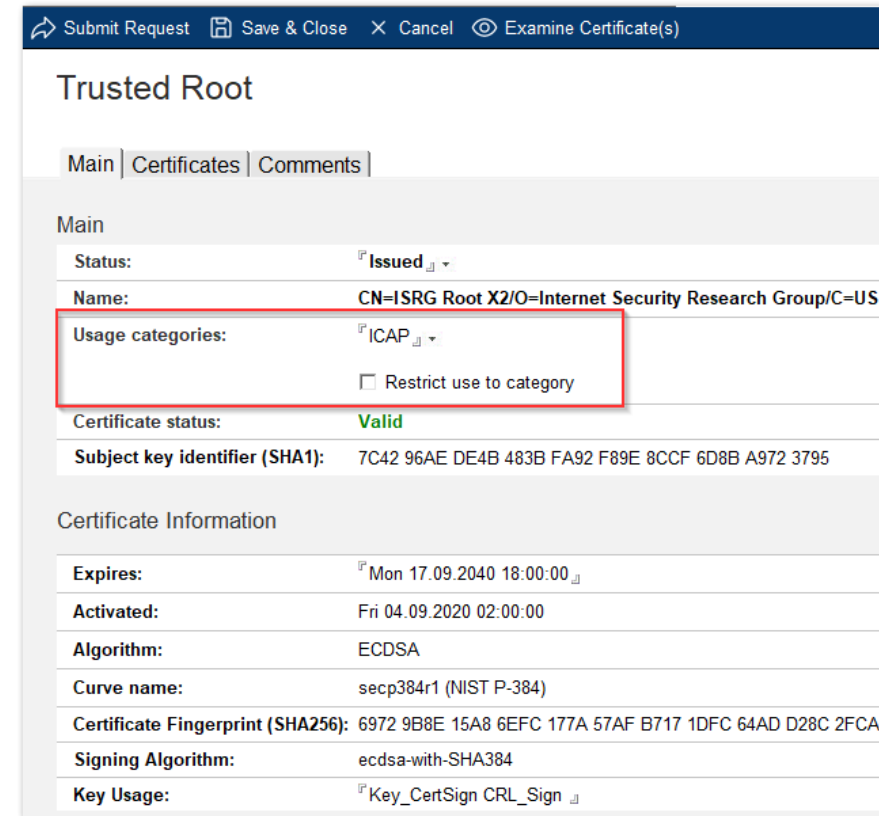
| Main                           |   |
|--------------------------------|---|
| Status:                        | Issued  |
| Name:                          | CN=ISRG Root X2/O=Internet Security Research Group/C=US |
| Usage categories:              |   |
| Certificate status:            | Valid   |
| Subject key identifier (SHA1): | 7C42 96AE DE4B 483B FA92 F89E 8CCF 6D8B A972 3795       |

| Certificate Information           |   |
|-----------------------------------|---|
| Expires:                          | Mon 17.09.2040 18:00:00                                       |
| Activated:                        | Fri 04.09.2020 02:00:00                                       |
| Algorithm:                        | ECDSA   |
| Curve name:                       | secp384r1 (NIST P-384)  |
| Certificate Fingerprint (SHA256): | 6972 9B8E 15A8 6EFC 177A 57AF B717 1DFC 64AD D28C 2FCA 8CF1 5 |
| Signing Algorithm:                | ecdsa-with-SHA384   |
| Key Usage:                        | Key_CertSign CRL_Sign   |

# Domino 12.0.2 Trusted Roots

- New certificate categories to assign trusted roots to applications like **ICAP** and **OIDC**
  - Can be used to restrict root certificates to a specific use cases
- Additional certificate details added
  - Curve name, SHA256 Fingerprint, Key usage, ...
- Easier to navigate view with categories
  - Hierarchical certificate information adapted from Domino directory, was confusing



Submit Request Save & Close X Cancel Examine Certificate(s)

## Trusted Root

Main Certificates Comments

Main

Status: Issued

Name: CN=ISRG Root X2/O=Internet Security Research Group/C=US

Usage categories: ICAP

☐ Restrict use to category

Certificate status: Valid

Subject key identifier (SHA1): 7C42 96AE DE4B 483B FA92 F89E 8CCF 6D8B A972 3795

### Certificate Information

Expires: Mon 17.09.2040 18:00:00

Activated: Fri 04.09.2020 02:00:00

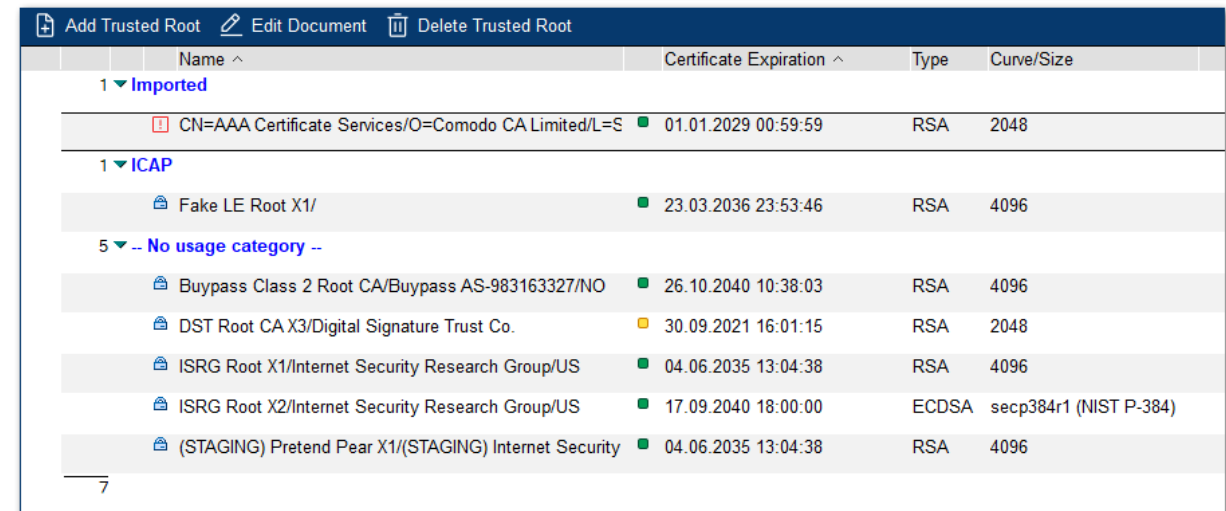
Algorithm: ECDSA

Curve name: secp384r1 (NIST P-384)

Certificate Fingerprint (SHA256): 6972 9B8E 15A8 6EFC 177A 57AF B717 1DFC 64AD D28C 2FCA

Signing Algorithm: ecdsa-with-SHA384

Key Usage: Key\_CertSign CRL\_Sign



| Name ^  | Certificate Expiration ^ | Type  | Curve/Size             |
|---|--------------------------|-------|------------------------|
| 1 Imported  |                          |       |                        |
| CN=AAA Certificate Services/O=Comodo CA Limited/L=S   | 01.01.2029 00:59:59      | RSA   | 2048                   |
| 1 ICAP  |                          |       |                        |
| Fake LE Root X1/                                      | 23.03.2036 23:53:46      | RSA   | 4096                   |
| 5 No usage category --                                |                          |       |                        |
| Buypass Class 2 Root CA/Buypass AS-983163327/NO       | 26.10.2040 10:38:03      | RSA   | 4096                   |
| DST Root CA X3/Digital Signature Trust Co.            | 30.09.2021 16:01:15      | RSA   | 2048                   |
| ISRG Root X1/Internet Security Research Group/US      | 04.06.2035 13:04:38      | RSA   | 4096                   |
| ISRG Root X2/Internet Security Research Group/US      | 17.09.2040 18:00:00      | ECDSA | secp384r1 (NIST P-384) |
| (STAGING) Pretend Pear X1/(STAGING) Internet Security | 04.06.2035 13:04:38      | RSA   | 4096                   |

# Domino 12.0.2 cerstore.nsf

- Automatically created on new servers with **One Touch Setup**
  - First server in a domain is always the **certmgr** server when setup with One Touch Setup
  - Additional servers replicate **certstore.nsf** from their setup server when setup with One Touch Setup
- Default process interval is now **2 seconds** instead of **30 seconds**
  - Important for remote request mode (like JConsole)



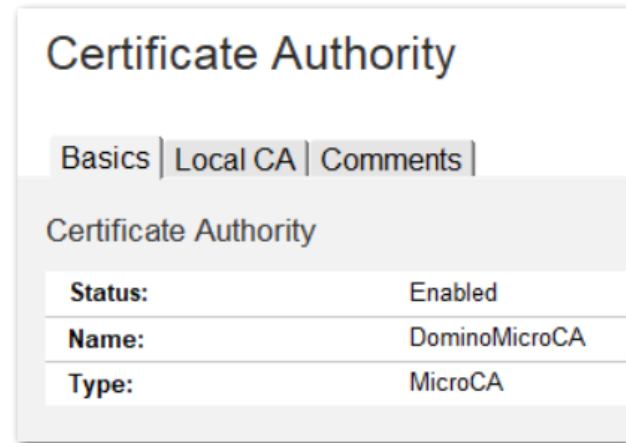
# HCLSoftware

Domino Micro CA



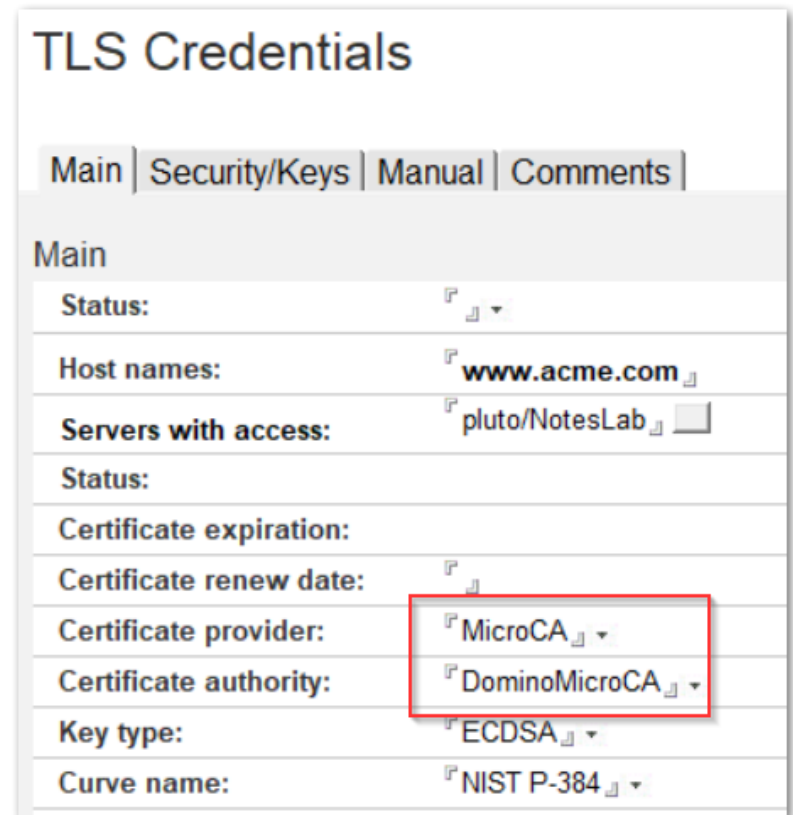
# Simple internal Micro CA

- If Let's Encrypt nor an internal CA is available ...
- Or you want a simple local CA for test or first server setup
- Domino 12.0.1 introduced a simple “**Micro CA**”
  - Managed by CertMgr
  - Available via One-touch setup for the first server in the domain
  - Or directly from **certstore.nsf** at any time issuing a certificate from the local CA
- Not a full CA – Only intended for testing & setup!!



The screenshot shows the 'Certificate Authority' configuration page. It has three tabs: 'Basics', 'Local CA', and 'Comments'. The 'Basics' tab is selected. Below the tabs, the page title is 'Certificate Authority'. There is a table with three rows: 'Status' set to 'Enabled', 'Name' set to 'DominoMicroCA', and 'Type' set to 'MicroCA'.

| Certificate Authority |               |
|-----------------------|---------------|
| Status:               | Enabled       |
| Name:                 | DominoMicroCA |
| Type:                 | MicroCA       |

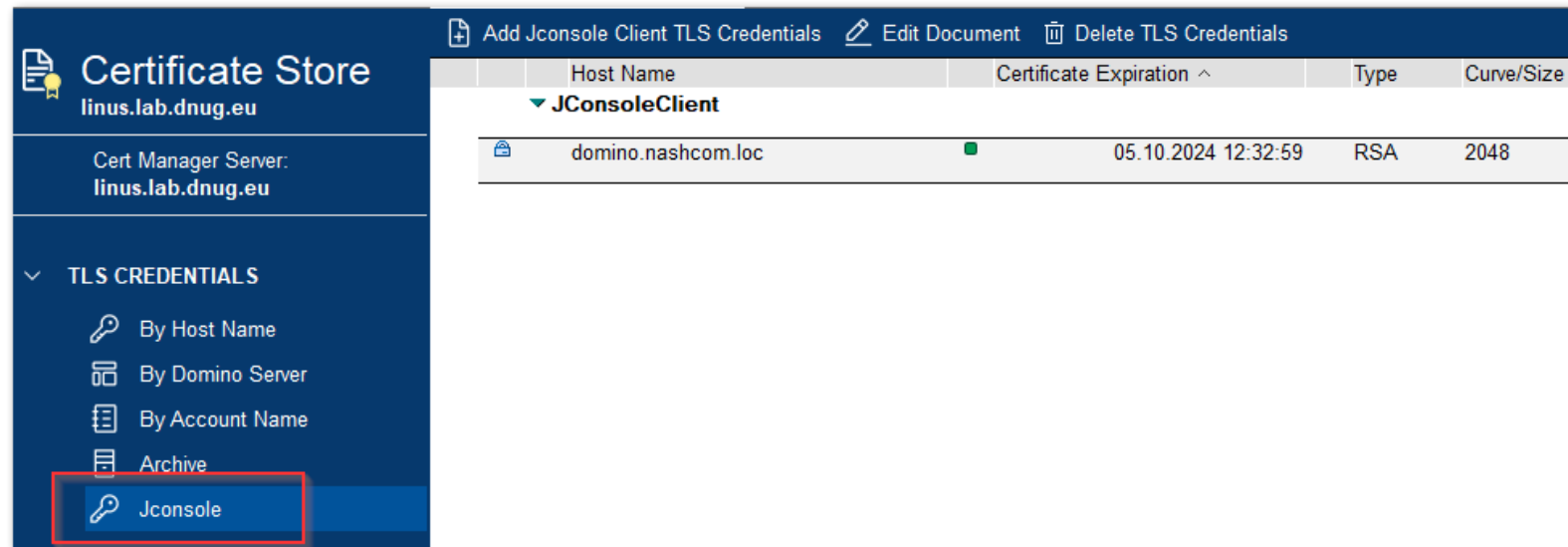


The screenshot shows the 'TLS Credentials' configuration page. It has four tabs: 'Main', 'Security/Keys', 'Manual', and 'Comments'. The 'Main' tab is selected. Below the tabs, the page title is 'Main'. There is a table with several rows: 'Status' (dropdown), 'Host names' (text field with 'www.acme.com'), 'Servers with access' (text field with 'pluto/NotesLab' and a button), 'Status' (text field), 'Certificate expiration' (text field), 'Certificate renewal date' (text field), 'Certificate provider' (dropdown with 'MicroCA' selected), 'Certificate authority' (dropdown with 'DominoMicroCA' selected), 'Key type' (dropdown with 'ECDSA' selected), and 'Curve name' (dropdown with 'NIST P-384' selected). A red rectangle highlights the 'Certificate provider' and 'Certificate authority' dropdowns.

| Main                      |                            |
|---------------------------|----------------------------|
| Status:                   | [dropdown]                 |
| Host names:               | [www.acme.com]             |
| Servers with access:      | [pluto/NotesLab] [button]  |
| Status:                   | [text field]               |
| Certificate expiration:   | [text field]               |
| Certificate renewal date: | [text field]               |
| Certificate provider:     | [MicroCA] [dropdown]       |
| Certificate authority:    | [DominoMicroCA] [dropdown] |
| Key type:                 | [ECDSA] [dropdown]         |
| Curve name:               | [NIST P-384] [dropdown]    |

# Domino 12.0.2 Micro CA

- New created MicroCAs are **10 years** valid instead of **1 year**
  - **Now supports exportable private keys!**
    - Can be used outside Domino if created exportable
- MicroCA is also used for **JConsole** certificates
- Remote request mode
  - Server posts request into cerstore.nsf on CertMgr server
  - CertMgr Server process request
  - Remote server polls until key & certificate is created



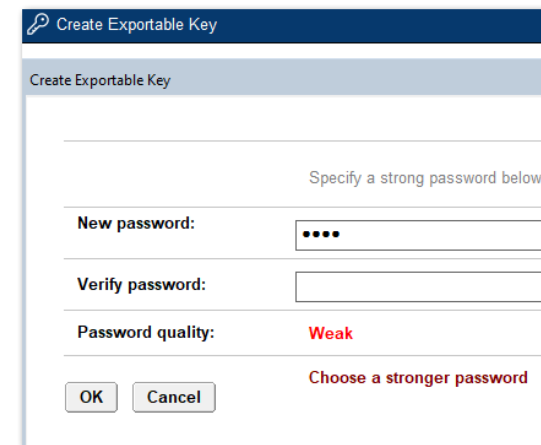
# Domino 12.0.2 JConsole Certs

- New JConsole certificates issued by Micro CA for the server's hostname
  - Server cert → **10 years**
  - Client cert → **2 years**
- Server certs are always created via command-line
  - `certmgmt create mca controller myhost.example.com`
  - `certmgmt create mca console myhost.mydomain.com`
- Client certs can be also created via UI
  - Requires exportable key with password
  - Keys are always RSA 2048 (Java 1.8 only supports RSA)
- Note: New servers automatically create JConsole TLS credentials when setup via OneTouch Setup

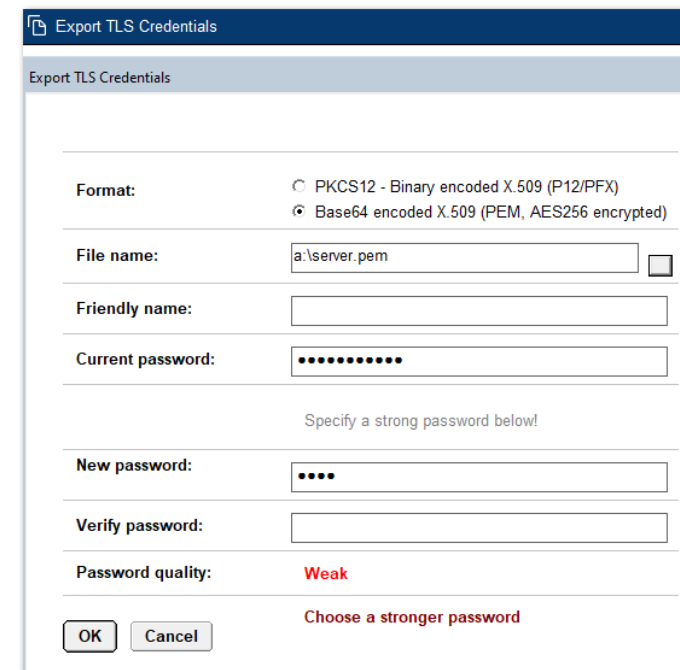
| TLS Credentials                          |                         |
|--|-------------------------|
| Main   Security/Keys   Manual   Comments |                         |
| Main                                     |                         |
| Status:                                  | Issued                  |
| Certificate Type:                        | JConsoleClient          |
| Host names:                              | domino.nashcom.loc      |
| Status:                                  | Valid                   |
| Certificate expiration:                  | Sat 05.10.2024 12:32:59 |
| Certificate provider:                    | MicroCA                 |
| Certificate authority:                   | Domino JConsole CA      |
| Key type:                                | RSA                     |
| Key size:                                | 2048                    |
| Certificate Attributes                   |                         |
| Common Name (CN):                        |                         |
| Organization:                            | dnug-lab                |
| Country:                                 | DE                      |

# Domino 12.0.1+ Exportable Private Keys

- Keys create or imported into certstore.nsf are encrypted for CertMgr server and servers listed in: “**Server with access**”
- By design those keys cannot be exported!
- But you can create an **exportable key**
  - Stored in encrypted PEM format in a separate field
  - Always encrypted with a password with reasonable entropy
- Full export & import dialog for **PEM**, **PKCS12** & **KYR** (import only)
- All import functionality provides
  - Certificate chain auto sorting and filtering
  - Certificate chain completion from trusted roots (even multi level)



The 'Create Exportable Key' dialog box is shown. It has a title bar with a magnifying glass icon and the text 'Create Exportable Key'. Below the title bar is a subtitle 'Create Exportable Key'. The main area contains a section titled 'Specify a strong password below!'. It includes three input fields: 'New password:' with masked characters '....', 'Verify password:' which is empty, and 'Password quality:' which displays 'Weak' in red text. Below these fields are 'OK' and 'Cancel' buttons. A red message 'Choose a stronger password' is displayed at the bottom right.



The 'Export TLS Credentials' dialog box is shown. It has a title bar with a document icon and the text 'Export TLS Credentials'. Below the title bar is a subtitle 'Export TLS Credentials'. The main area contains a section titled 'Specify a strong password below!'. It includes a 'Format:' section with two radio buttons: 'PKCS12 - Binary encoded X.509 (P12/PFX)' and 'Base64 encoded X.509 (PEM, AES256 encrypted)', with the latter selected. Below this is a 'File name:' field with the text 'a:\server.pem' and a checkbox. There is also a 'Friendly name:' field. Below these is a 'Current password:' field with masked characters '.....'. At the bottom, there are 'New password:' and 'Verify password:' fields, both with masked characters. The 'Password quality:' section displays 'Weak' in red text. 'OK' and 'Cancel' buttons are at the bottom left, and a red message 'Choose a stronger password' is at the bottom right.

# HCLSoftware

CertMgr Certificate  
URL Health Check



# CertMgr Certificate URL Health Check

- Can be configured in each **TLS Credentials doc** to check certificate health on servers
- Supports all standard TLS connections (HTTPS, LDAPS, IMAPS, POP3S, ...)
  - Does not support SMTP STARTTLS which starts the connection unencrypted on port 25
- Check performed once per day
- Manual check via: **tell certmgr check**
- Can send daily notification e-mail and writes statistics

| Administration        |  |
|-----------------------|--|
| Health check URLs:    | linus.lab.dnug.eu<br>traveler.lab.dnug.eu<br>www.lab.dnug.eu<br>icap.lab.dnug.eu:1344                                      |
| Health check options: | <input checked="" type="checkbox"/> Enabled<br><input checked="" type="checkbox"/> Use trusted roots from Domino Directory |

| Global Settings                  |                            |
|----------------------------------|----------------------------|
| Basics                           | Comments                   |
| Global Parameters                |                            |
| Admin Server:                    | linus.lab.dnug.eu/dnug-lab |
| Health Check notification email: | Admin/dnug-lab             |

# Mail & Log Example



**CertMgr URL Health Check - Failures: 1, Warnings: 3**  
**linus.lab.dnug.eu** to: Admin

## Certificate failures (1)

**icap.lab.dnug.eu:1344** - Failed to connect to icap.lab.dnug.eu port 1344 after 12 ms: Connection refused

## Certificate expiration warnings (3)

**linus.lab.dnug.eu** (37.4 days)

**traveler.lab.dnug.eu** (37.4 days)

**www.lab.dnug.eu** (37.4 days)

```
tell certmgr check
> [003345:000002-00007FCE9E68EC00] CertMgr: Checking ..
[003345:000006-00007FCE8A1FB700] Checking for requests ..
[003345:000006-00007FCE8A1FB700] 10/06/2022 05:18:59 CertMgr: Info: Health Check - Green: 7 Yellow: 0 Red: 1
[003345:000006-00007FCE8A1FB700] 10/06/2022 05:18:59 CertMgr: Warning - URL Health Check [linus.lab.dnug.eu], certificate will expire 11/12/2022 15:07:25 (37.4 days)
[003345:000006-00007FCE8A1FB700] 10/06/2022 05:18:59 CertMgr: Warning - URL Health Check [traveler.lab.dnug.eu], certificate will expire 11/12/2022 15:07:25 (37.4 days)
[003345:000006-00007FCE8A1FB700] 10/06/2022 05:18:59 CertMgr: Warning - URL Health Check [www.lab.dnug.eu], certificate will expire 11/12/2022 15:07:25 (37.4 days)
[003345:000006-00007FCE8A1FB700] 10/06/2022 05:18:59 CertMgr: Failure - URL Health Check [icap.lab.dnug.eu:1344] : Failed to connect to icap.lab.dnug.eu port 1344 after 12 ms
: Connection refused
[003345:000006-00007FCE8A1FB700] 10/06/2022 05:18:59 CertMgr: Info: URL Health Check - Green: 0 Yellow: 3 Red: 1
[003298:000017-00007F5456D35700] 10/06/2022 05:18:59 Router: Message 001D342D delivered to Admin/dnug-lab
```



# New Health Check URL Statistics

- Can be used to generate custom notifications via event monitoring
- Read/Yellow/Green status similar to CertStatus

```
show stat certmgr.*
[003130:000009-00007FB9BC4FC700] CertMgr.CertStatus = Red
[003130:000009-00007FB9BC4FC700] CertMgr.CertStatus.Green = 7
[003130:000009-00007FB9BC4FC700] CertMgr.CertStatus.Red = 1
[003130:000009-00007FB9BC4FC700] CertMgr.CertStatus.Yellow = 0
[003130:000009-00007FB9BC4FC700] CertMgr.HealthCheckURL.CheckTime.Last = 10/06/2022 05:18:59 GMT
[003130:000009-00007FB9BC4FC700] CertMgr.HealthCheckURL.CheckTime.Next = 10/07/2022 05:18:59 GMT
[003130:000009-00007FB9BC4FC700] CertMgr.HealthCheckURL.IntervalHours = 24
[003130:000009-00007FB9BC4FC700] CertMgr.HealthCheckURL.Status.Green = 0
[003130:000009-00007FB9BC4FC700] CertMgr.HealthCheckURL.Status.Red = 1
[003130:000009-00007FB9BC4FC700] CertMgr.HealthCheckURL.Status.Yellow = 3
[003130:000009-00007FB9BC4FC700] CertMgr.Status = Red
[003130:000009-00007FB9BC4FC700] 11 statistics found
```

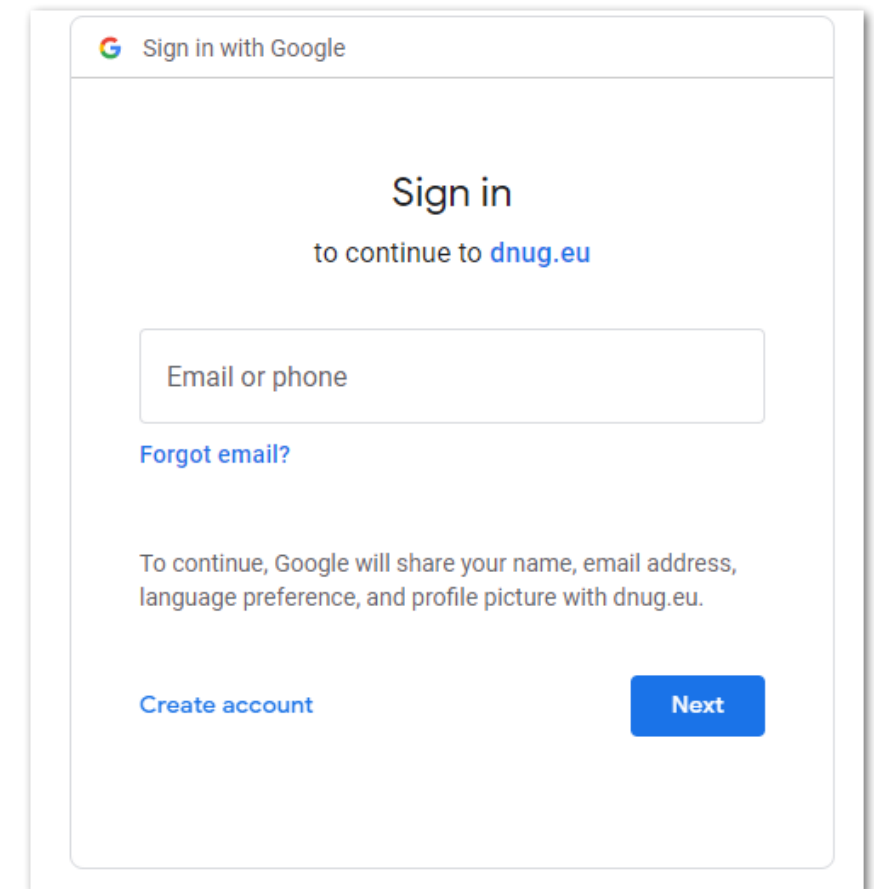
# HCLSoftware

OpenID /OIDC Support



# OpenID Connect 1.0 (OIDC) Authentication

- Allows to use OpenID Connect 1.0 (OIDC) compliant 3<sup>rd</sup> party IdPs for authentication
  - Check <https://openid.net> for details
  - “Similar” to **SAML** but **easier to configure & more modern**
- **Tested providers**
  - KeyCloak
  - Google
  - Yahoo
  - Microsoft Azure AD
- **Untested providers (Any volunteers?)**
  - Microsoft ADFS 2019+ (On-prem)
  - Okta (On-prem)
  - PingFederate (On-prem)
  - Salesforce (per customer)



# Providers known to not work (and why)

- **Apple-ID**
  - Doesn't support **client\_secret\_basic**
  - Apple uses a custom variant of **private\_key\_jwt** authentication
- **AWS IAM Identity Center (successor to AWS Single Sign On)**
  - Does not support the Authorization Code Flow with PKCE
- **Facebook**
  - Does not currently support the authorization code flow and does not expose a token endpoint
- **GitHub**
  - Supports OAuth, but no well-known endpoint and will not return an **id\_token**
- **Twitter**
  - Does not support OIDC

# OIDC / OpenID Authentication

- Enabled in internet site document
- Requires a OIDC document in **idpcat.nsf**
  - Provider needs to support the full OIDC standard and have a valid **.well-known/openid-configuration**
- **Requires end to end TLS encryption!**
  - In case of TLS termination on secure proxy, use separate TLS connection between proxy and Domino
  - Tip: Domino MicroCA can be used to issue certificates
  - Only 1 year valid, but auto renewed by CerMgr

Web Site DNUG LAB

Basics | Configuration | Domino Web Engine | Security

**TCP Authentication**

|                      |   |
|----------------------|---|
| Anonymous:           | <input type="radio"/> Yes <input checked="" type="radio"/> No |
| Name & password:     | <input type="radio"/> Yes <input checked="" type="radio"/> No |
| Redirect TCP to TLS: | <input checked="" type="radio"/> Yes <input type="radio"/> No |

**TLS Authentication**

|                     |   |
|---------------------|---|
| Anonymous:          | <input type="radio"/> Yes <input checked="" type="radio"/> No |
| Name & password:    | <input checked="" type="radio"/> Yes <input type="radio"/> No |
| Client certificate: | <input type="radio"/> Yes <input checked="" type="radio"/> No |
| Bearer token (JWT): | <input checked="" type="radio"/> Yes <input type="radio"/> No |

**TLS Options**

|                |                   |
|----------------|-------------------|
| Key file name: | linus.lab.dnug.eu |
|----------------|-------------------|

Edit OIDC Provider Cancel

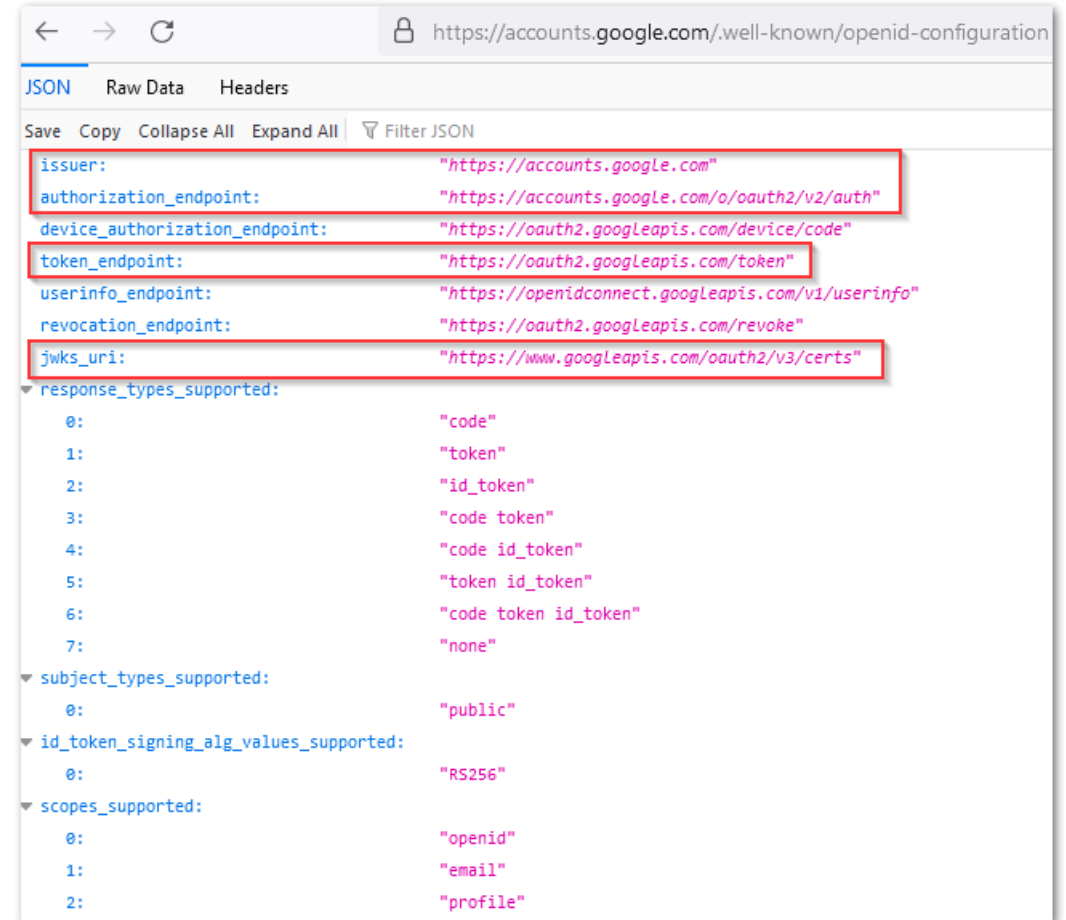
**OIDC Provider : Google**

**OIDC Provider Settings**

|  |  |
|--|--|
| Host names or addresses mapped to this site: | linus.lab.dnug.eu                              |
| Provider name:                               | Google   |
| Base URL:                                    | https://accounts.google.com                    |
| Trusted roots:                               | GlobalSign Root CA/Root CA/GlobalSign nv-sa/BE |
| Logging level:                               | Verbose  |

# Google OpenID Configuration

- <https://accounts.google.com/.well-known/openid-configuration>
- Required
  - issuer
  - authorization\_endpoint
  - token\_endpoint
  - jwks\_uri
- Only specify the URL without **/.well-known..**
  - The server always uses this standard location!



# Trusted Root Configuration

- Trusted root configuration is optional
  - Without trusted root the underlying LibCurl code uses **cacert.pem** in server's data directory
- Trusted root is selected from **certstore.nsf**
- Tip: Import trusted roots
  - `load certmgr -ImportRootFromUrl https://accounts.google.com/.well-known/openid-configuration OIDC`
    - Checks the remote site and creates a new draft trusted root document for OIDC use
    - If remote site does not send a trusted root, certificate chain is checked against Domino directory to auto complete the chain and add the trusted root into **certstore.nsf**
    - Trusted root needs to be verified in **certstore.nsf** before it can be used

# Trusted Root Validation

[Submit Request](#) [Save & Close](#) [Cancel](#) [Examine Certificate\(s\)](#) [✓ Mark trusted root validated](#)

## Trusted Root

[Main](#) [Certificates](#) [Comments](#)

Main

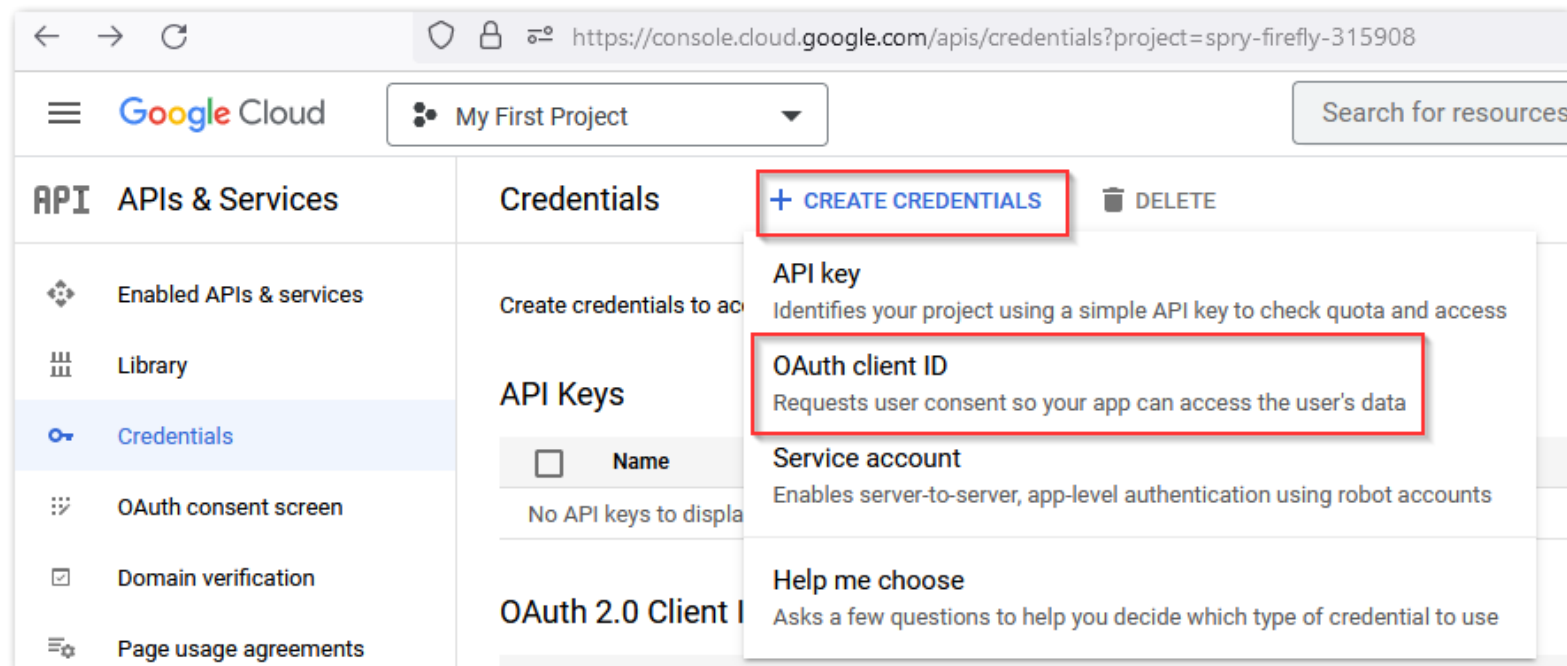
|                                |  |
|--------------------------------|--|
| Status:                        | 『 Pending Validation 』 ▾   |
| Name:                          | CN=GlobalSign Root CA/OU=Root CA/O=GlobalSign nv-sa/C=BE                         |
| Usage categories:              | 『 OIDC, ICAP 』 ▾<br><input checked="" type="checkbox"/> Restrict use to category |
| Certificate status:            | Valid  |
| Subject key identifier (SHA1): | 607B 661A 450D 97CA 8950 2F7D 04CD 34A8 FFFC FD4B                                |

|                 |   |   |                     |     |      |
|-----------------|---|---|---------------------|-----|------|
| 3 ▾ <b>OIDC</b> |   |   |                     |     |      |
| ❗               | DigiCert Global Root CA/www.digicert.com/DigiCert Inc/U | ■ | 10.11.2031 01:00:00 | RSA | 2048 |
| ❗               | DigiCert High Assurance EV Root CA/www.digicert.com/E   | ■ | 10.11.2031 01:00:00 | RSA | 2048 |
| ❗               | GlobalSign Root CA/Root CA/GlobalSign nv-sa/BE          | ■ | 28.01.2028 13:00:00 | RSA | 2048 |



# Google OIDC Configuration

- Documentation
  - <https://developers.google.com/identity/protocols/oauth2/openid-connect>
- Configuration
  - <https://console.cloud.google.com/apis/dashboard>



# Google OIDC Configuration

- Web Application
- Specify a name
- Set the URL
- Always Server URL + /names.nsf?OIDCLogin

[←](#) Create OAuth client ID

A client ID is used to identify a single app to Google's OAuth servers. If your app runs on multiple platforms, each will need its own client ID. See [Setting up OAuth 2.0](#) for more information. [Learn more](#) about OAuth client types.

Application type \*

Web application

Name \*

dnug-lab-web

The name of your OAuth 2.0 client. This name is only used to identify the client in the console and will not be shown to end users.

**i**

The domains of the URIs you add below will be automatically added to your [OAuth consent screen](#) as [authorized domains](#).

Authorized JavaScript origins **?**

For use with requests from a browser

+ ADD URI

Authorized redirect URIs **?**

For use with requests from a web server

URIs 1 \*

<https://linus.lab.dnug.eu/names.nsf?OIDCLogin>

+ ADD URI

Note: It may take 5 minutes to a few hours for settings to take effect

CREATE

CANCEL

# Google OIDC Configuration

- Generates
  - **Client ID**
  - **Client Secret** \*) sample secret already replaced
- OIDC Client ID + Secret need to be stored in **Notes.ini**
  - Notes.ini instead of form data because OpenID Support was a last minute addition based on feedback from EAP Forum
  - set config **OIDC\_LOGIN\_CLIENT\_ID**=990428096234-tiq585g98arppjhujpg64aj4hvru0j4d.apps.googleusercontent.com
  - set config **OIDC\_LOGIN\_CLIENT\_SECRET**=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
- set config **OIDC\_LOGIN\_ENABLE\_REDIRECT=1**
  - Enables login redirect for OIDC

## OAuth client created

The client ID and secret can always be accessed from Credentials in APIs & Services



OAuth access is restricted to the [test users](#) listed on your [OAuth consent screen](#)

Your Client ID

990428096234-tiq585g98arppjhujpg64aj4hvru0j4d.apps.googleusercontent.com



Your Client Secret

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



# OIDC map User Name

- Remote name is passed via **e-mail attribute**
- Add external e-mail addresses to the corresponding person document
- notes.ini **OIDC\_CUSTOM\_EMAIL\_CLAIM\_NAME** to use custom claim instead of "email" claim



The screenshot shows a user profile form for Daniel Nashed. At the top, the text "Daniel Nashed/NashCom/DE" and "nsh@nashcom.de@dnug-lab" is displayed. Below this is a tabbed interface with tabs for "Basics", "Work/Home", "Other", "Miscellaneous", "Certificates", "Roaming", and "Administration". The "Basics" tab is selected. Under the "Basics" tab, there is a section for personal information with fields for "First name" (Daniel), "Middle name" (empty), and "Last name" (Nashed). Below this is another section for "Basics" with fields for "User name" and "Alternate name". The "User name" field contains the text "Daniel Nashed/NashCom/DE" and "daniel.nashed@gmail.com", with the latter part highlighted by a red box. The "Alternate name" field is empty.

Daniel Nashed/NashCom/DE  
nsh@nashcom.de@dnug-lab

Basics | Work/Home | Other | Miscellaneous | Certificates | Roaming | Administration

First name: Daniel  
Middle name:  
Last name: Nashed

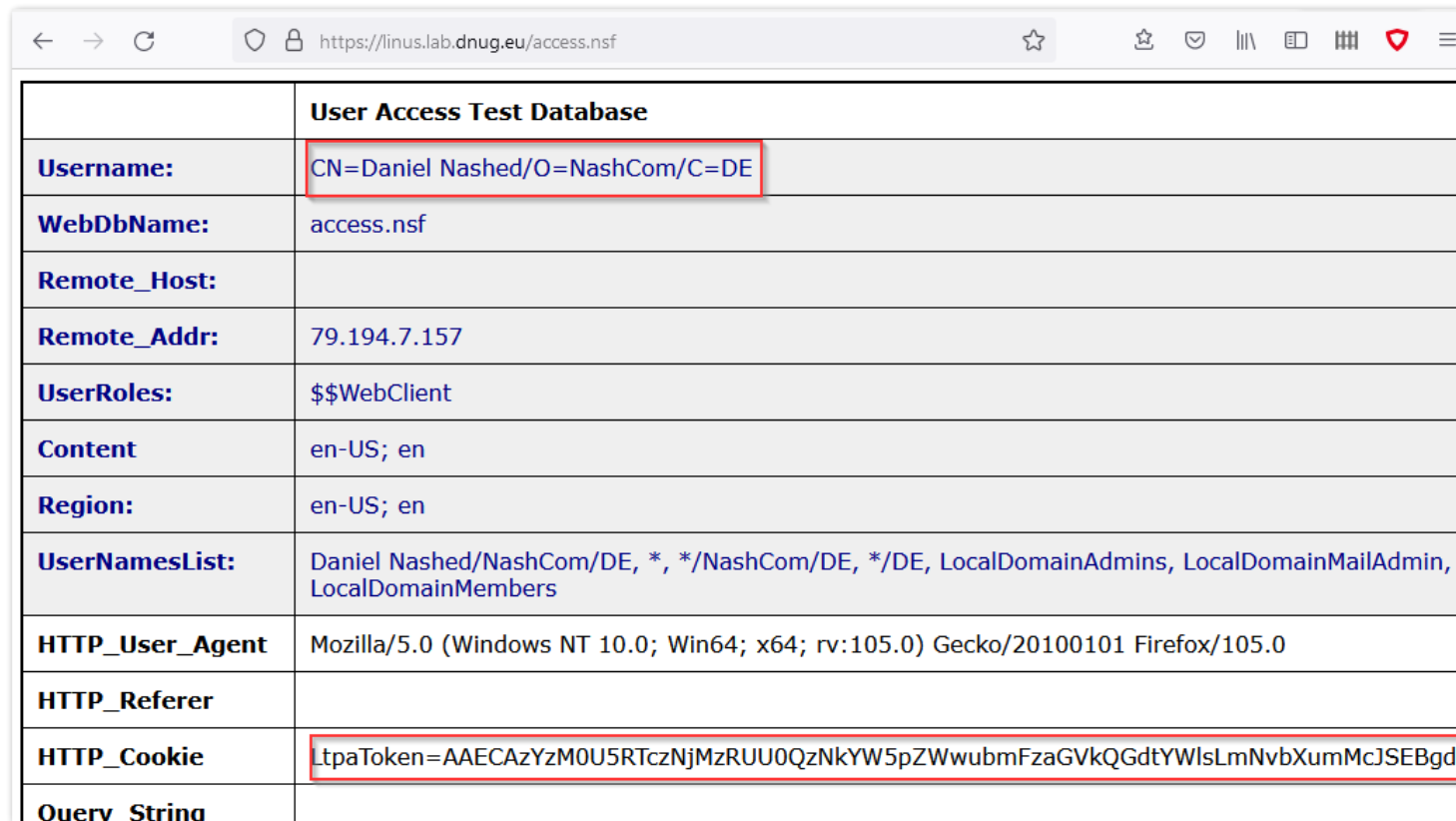
Basics

User name: Daniel Nashed/NashCom/DE  
daniel.nashed@gmail.com

Alternate name:

# Authenticated User Example

- User is mapped and first entry in **Fullname** field is used to build the **UserNamesList**
- LTPA SSO and single server sessions are supported



|                 | User Access Test Database  |
|-----------------|--|
| Username:       | CN=Daniel Nashed/O=NashCom/C=DE  |
| WebDbName:      | access.nsf   |
| Remote_Host:    |  |
| Remote_Addr:    | 79.194.7.157   |
| UserRoles:      | \$\$WebClient  |
| Content         | en-US; en  |
| Region:         | en-US; en  |
| UserNamesList:  | Daniel Nashed/NashCom/DE, *, */NashCom/DE, */DE, LocalDomainAdmins, LocalDomainMailAdmin, LocalDomainMembers |
| HTTP_User_Agent | Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:105.0) Gecko/20100101 Firefox/105.0                             |
| HTTP_Referer    |  |
| HTTP_Cookie     | LtpaToken=AAECAzYzM0U5RTczNjMzRUU0QzNkYW5pZWwubmFzaGVkQGdtYWlsLmNvbXumMcJSEBgd                               |
| Query String    |  |

# HCLSoftware

## Sender Policy Framework (SPF)



# Sender Policy Framework (SPF)

- RFC 7208 - Sender Policy Framework (SPF) for Authorizing Use of Domains in Email, Version 1
  - <https://datatracker.ietf.org/doc/html/rfc7208>
- Defines which host are **allowed to send mails for a domain**
- **DNS TXT record** for a domain or sub-domain with flexible rule set
- Example:  
`host -t txt lab.dnug.eu -> lab.dnug.eu descriptive text "v=spf1 mx ~all"`
  - Only allows domain's defined MX record hosts to send mail
- More complex example **dnug.de**  
`v=spf1 mx  
a:domino.dnug.de ip4:87.230.23.16  
include:spf.nl2go.com include:mail.zendesk.com include:spf.ce.cloud-y.com  
-all`

# SPF Syntax

- [http://www.open-spf.org/SPF\\_Record\\_Syntax](http://www.open-spf.org/SPF_Record_Syntax)

## Mechanisms

Mechanisms can be prefixed with one of four qualifiers:

"+" Pass  
"\_" Fail  
"~" SoftFail  
"?" Neutral

If a mechanism results in a hit, its qualifier value is used. The default qualifier is "+", i.e. "Pass". For example:

```
"v=spf1 -all"  
  
"v=spf1 a -all"  
  
"v=spf1 a mx -all"  
  
"v=spf1 +a +mx -all"
```

### The "ip4" mechanism [\(edit\)](#)

```
ip4:<ip4-address>  
ip4:<ip4-network>/<prefix-length>
```

The argument to the "ip4:" mechanism is an IPv4 network range. If no *prefix-length*

Examples:

```
"v=spf1 ip4:192.168.0.1/16 -all"
```

Allow any IP address between 192.168.0.1 and 192.168.255.255.

### The "include" mechanism [\(edit\)](#)

```
include:<domain>
```

The specified *domain* is searched for a match. If the lookup does not return a match or an error, processing reject based on a *PermError*.

Examples:

In the following example, the client IP is 1.2.3.4 and the *current-domain* is example.com.

```
"v=spf1 include:example.com -all"
```

If example.com has no SPF record, the result is *PermError*.

Suppose example.com's SPF record were "v=spf1 a -all".

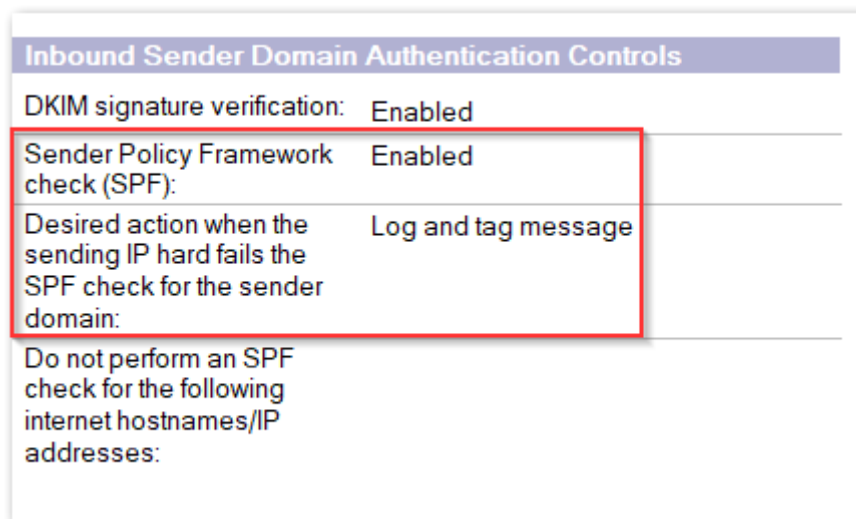
Look up the A record for example.com. If it matches 1.2.3.4, return *Pass*.

If there is no match, other than the included domain's "-all", the include as a whole fails to match; t



# SPF Inbound Support in Domino 12.0.2

- SPF checks can be enabled in server configuration
- Can be used to deliver mail to SPAM folder
  - Not helpful in all customer scenarios
  - But also adds a SPF field to the message leveraged in other applications like Nash!Com SpamGeek
- Enable via Config Doc: **Router/SMTP / SMTP Inbound Controls**
  - Select: Log and tag message → Adds a new field **Received\_SPF** to inbound SMTP messages



| Inbound Sender Domain Authentication Controls                                      |                     |
|--|---------------------|
| DKIM signature verification:   | Enabled             |
| Sender Policy Framework check (SPF):   | Enabled             |
| Desired action when the sending IP hard fails the SPF check for the sender domain: | Log and tag message |
| Do not perform an SPF check for the following internet hostnames/IP addresses:     |                     |

# Enable Inbound SPF Checking

| Inbound Sender Domain Authentication Controls                                      |                     |
|--|---------------------|
| DKIM signature verification:   | Enabled             |
| Sender Policy Framework check (SPF):   | Enabled             |
| Desired action when the sending IP hard fails the SPF check for the sender domain: | Log and tag message |
| Do not perform an SPF check for the following internet hostnames/IP addresses:     |                     |

- Config Doc: **Router/SMTP / SMTP Inbound Controls**
  - Select: **Log and tag message**
  - Adds a field **Received\_SPF** to inbound SMTP messages
- **Received\_SPF** field
  - contains status + additional information

Field Name: Received\_SPF

Data Type: RFC822 Text

"pass (notes.nashcom.de: domain of pnp-hcl.com designates 3.226.151.152 as permitted sender) client-ip=3.226.151.152; envelope-from=john.doe@pnp-hcl.com; helo=smtp1.mail.cwp.pnp-hcl.com;"

# HCLSoftware

## DKIM

Domain Keys Identified Mail



# Domain Keys Identified Mail (DKIM)

- Allows senders to sign parts of the message to allow a receiving server to verify the signature of a published public key in DNS
- **RFC 6376** - DomainKeys Identified Mail (DKIM) Signatures
  - <https://datatracker.ietf.org/doc/html/rfc6376>
- Signing keys per domain stored in DNS TXT Records
- Example: `host -t txt ed20220604._domainkey.lab.dnug.eu`  
  
`"v=DKIM1; k=ed25519; p=P+qCLYFRh7QmmqZV4ossGeZTmLyrqI8/nU0fZHd52v0="`
- There can be multiple public keys with a lookup by a “**selector**”
  - Most environments still use RSA. Domino supports more modern **Ed25519** keys – in parallel (dual signature)
  - There can be more selectors to define keys. Also useful for **key rollover**

# DKIM Signature

- Signature is calculated based on defined fields of the message
- DKIM header added to the message
- Receiving server
  - Finds the selector in the header
  - Queries the DNS TXT record for selector/domain
  - Verifies message using the public key
- Example: mail from **admin@lab.dnug.eu**
- DKIM-Signature: v=1; a=ed25519-sha256; c=relaxed/relaxed; d=lab.dnug.eu;  
s=ed20220604; t=1654333615;  
bh=vUKg8XaDsgHuWYIPJChU9IFoYcm+6Bi7pkbXtoa4qo=;  
h=To:Cc:MIME-Version:Subject:Message-ID:From:Date:Content-Type;  
b=oTenSwxCs58gqMSI0iVuDZCN4zf1IV5f6kN1qv4MoPZ8y4MyABgb5nrrAUOANOWYb  
Ef6TcaE/kYihPS5gj0FAA==

# Domino 12.0.1 - Enable outbound DKIM

- Run console command to create a DKIM key
  - `keymgmt create DKIM lab.dnug.eu ed20220604 ed25519`
- Run console command to create a file containing the DNS TXT record
  - `keymgmt export DKIM DNS lab.dnug.eu ed20220604 lab_dnug_eu_ed20220604.txt`
- Create a DNS TXT record for `ed20220604._domainkey.lab.dnug.eu`
- Define DKIM key for the domain, enable DKIM outbound signing and restart router
  - `set config DKIM_KEY_lab.dnug.eu=ed20220604`
  - `set config RouterDKIMSigning=1`
  - Restart task router

# Domino 12.0.2 - Enable inbound DKIM

- Enable via Config Doc: **Router/SMTP / SMTP Inbound Controls**
- Adds new field “**DKIM\_Signature**” to inbound SMTP message

| Inbound Sender Domain Authentication Controls                                      |                     |
|--|---------------------|
| DKIM signature verification:   | Enabled             |
| Sender Policy Framework check (SPF):   | Enabled             |
| Desired action when the sending IP hard fails the SPF check for the sender domain: | Log and tag message |
| Do not perform an SPF check for the following internet hostnames/IP addresses:     |                     |

Field Name: DKIM\_Signature  
Data Type: RFC822 Text  
"v=1; a=ed25519-sha256; c=relaxed/relaxed; d=ppn-hcl.com; s=ed10122021; t=1664916270; bh=cKcBERK1YNs97d4zgyrEevIRwTZx9kuELxhiMDtGxSw=; h=In-Reply-To:References-To:Cc:MIME-Version:Subject:From:Message-ID: Date:Content-Type; b=Em1DGn9odhI34JiXsTvIEA/YZFTQ6vLkmuG1LJJKuvNkw955iJXy8VKF4tWqX16LZ 1Prwh/1JRORFb9mzBlaCQ=="

# Domino 12.0.2 – DKIM & SPF Status

- Field **Authentication\_Results** contains result from DKIM and SPF
- Field Name: Authentication\_Results  
Authentication\_Results: notes.lab 1; **spf=pass** smtp.mailfrom=nsh@notes.lab (sender IP 1.2.3.4); **dkim=pass** header.s=09302021 header.d=notes.lab; **dkim=pass** header.s=ed10122021 header.d=notes.lab
- Reference
  - <https://www.rfc-editor.org/rfc/rfc7001>
- Currently only two options are available
  - Log and Tag
  - Deliver to Junk
- External tools like Nash!Com SpamGeek can leverage the new field



# HCLSoftware

## CScan - Antivirus

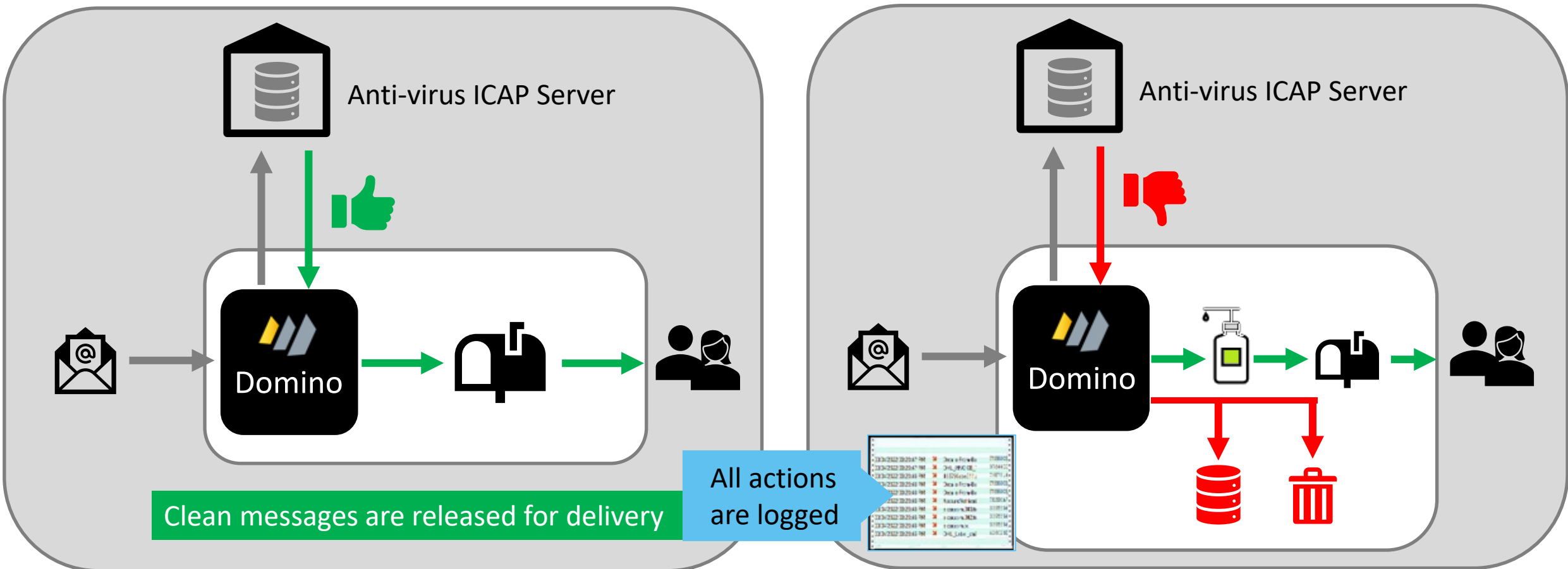
ICAP (Internet Content  
Adaptation Protocol)



# CScan – Antivirus leveraging ICAP protocol

- Invented for **Proxy security**, but can be used for antivirus checking attachments as well

Virus Detected



# Internet Content Adaptation Protocol (ICAP)

- **RFC 2507** - Internet Content Adaptation Protocol (**ICAP**)
  - <https://datatracker.ietf.org/doc/html/rfc3507>
- Domino 12.0.2 natively implements the ICAP protocol and leverages it for attachment scanning
- Support for Windows 64 / Linux 64 in Domino 12.0.2
- New “**mailscan**” server task is integrated into **mail router message flow**

# ICAP Providers

- Trend Micro™ Web Security
  - [https://www.trendmicro.com/en\\_us/business/products/user-protection/sps/web-security.html](https://www.trendmicro.com/en_us/business/products/user-protection/sps/web-security.html)
- McAfee™ Web Gateway
  - <https://www.mcafee.com/enterprise/en-us/downloads/trials/web-protection-evaluation.html>
- For testing only
  - **C-ICAP open source project** using ClamAV in the back-end (<https://c-icap.sourceforge.net/> )
    - Ulrich Krause put together a detailed step by step setup documentation <https://www.eknori.de/2022-05-31/domino-12-0-2-eap-cd-1-clamav-icap/>
  - **ICAP mock server** available until EAP4
    - Can be copied from EAP4 -- A simple internal testing tool HCL shared during early beta
- If you have other **ICAP** solutions in place, I would like to hear from you!

# Domino 12.0.2 Mail Flow Content Scan

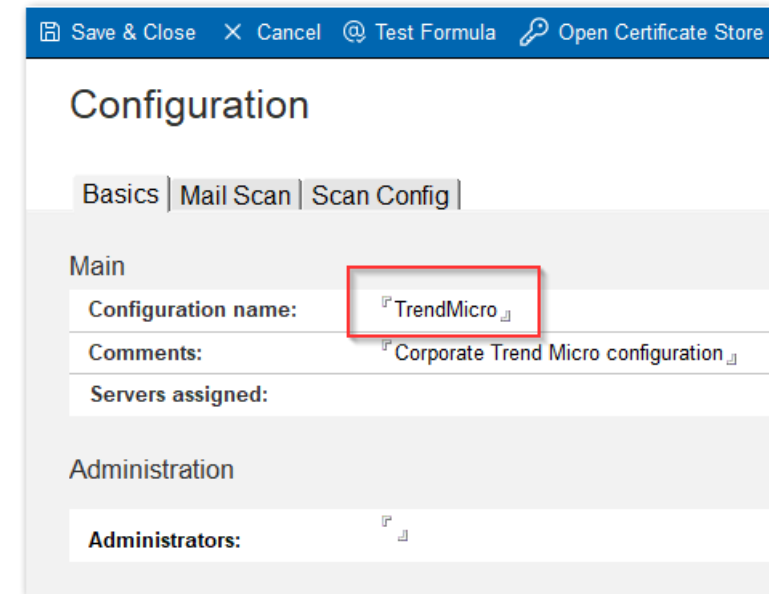
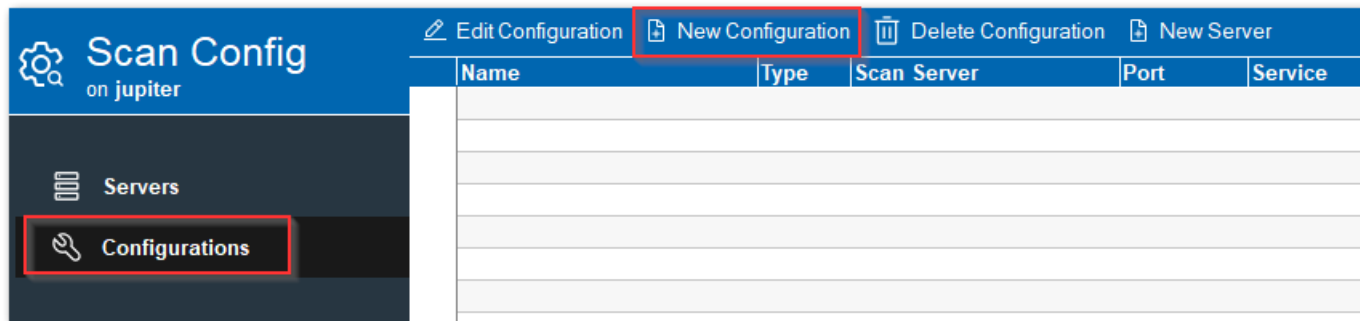
- Virus scanning for mail flow (mail router integration)
- Main components
  - **cscancfg.nsf**
    - Domain wide database for server configuration and status
  - **cscanlog.nsf**
    - Per server database to log virus events
  - **cscanquarantine.nsf**
    - Per server database to store quarantined message data
  - **mailscan** servertask integrated with mail router flow

# Configuration Flow

- **Load mailscan** creates the domain wide **cscancfg.nsf** configuration database
  - Tries to pull a replica from admin server if started on another server already
  - Creates a replica on admin server for other servers to replicate
- Once create open **csscancfg.nsf** to create a ICAP configuration
- Create server configuration with assigned ICAP configuration per server
- Finally load **mailscan** to validate the configuration by connecting to the ICAP server

# Create new Configuration

- Create new configuration document first
- Specify an unique configuration name
  - Cannot be changed once servers are assigned
  - Protected against deletion



# Specify Mail Scan Settings

- Virus detection, Quarantine and Log Options are predefined
- Can be changed based on customer needs
- Log all attachments only makes sense in test environments
- Mail Tag for Notifications should be set
  - Settings are optional, but should be set
  - Used in conjunction with “**Scan and Log Options**”

The screenshot shows the 'Configuration' window with the 'Mail Scan' tab selected. The 'Scan and Log Options' section is highlighted with a red box, and the 'Mail Tag for Notification' section is also highlighted with a red box.

| Scan and Log Options   |                                     |
|------------------------|-------------------------------------|
| Virus detected action: | Discard message with notification ▼ |
| Quarantine action:     | Quarantine original message ▼       |
| Message log option:    | Log attachments with viruses only ▼ |
| Log database:          | cscanlog.nsf                        |
| Quarantine database:   | cscanquarantine.nsf                 |
| Log retention (days):  | 40                                  |
| Quarantine (days):     | 40                                  |

| Mail Tag for Notification         |  |
|-----------------------------------|--|
| Subject prefix scanned:           | cscan: processed -                     |
| Subject prefix virus found:       | cscan: virus found -                   |
| Subject prefix message discarded: | cscan: virus found & message blocked - |
| Virus view icon:                  | 24                                     |
| Virus attachment text:            | Virus detected, attachment removed!    |
| Body text message discarded:      |  |

**cscan Notification**

Message blocked due to virus!



# Specify Mail Scan Settings

- **Get ICAP Configuration from ICAP Admin**
  - ICAP server name should be a DNS name!
  - ICAP standard port is often **1344** or **11344** for TLS
  - The port could vary depending on ICAP server
  - ICAP Service name needs to be specified
    - If ICAP server does not require a service name, specify any name
    - Trend Micro Web Gateway uses “**Interscan**”
- Specify optional “Virus name formula”
  - Formula is executed on result document and depends on headers returned by ICAP vendor

The screenshot shows the 'Configuration' window with the 'Mail Scan' tab selected. The 'Scan Configuration' section is highlighted with a red box. The settings are as follows:

| Field  | Value  |
|--|--|
| Scan protocol:   | ICAP   |
| Maximum scan size (MB):                                | 100.0  |
| ICAP server name (DNS):                                | tms.hcl.loc  |
| ICAP TLS server port:<br>(All connections require TLS) | 11344  |
| ICAP service name:                                     | Interscan  |
| ICAP preview:  | <input type="checkbox"/> Enable ICAP Preview         |
| Virus name formula:                                    | @Trim (@Right (ICAP_ResponseHeaders; "X-Virus-ID:")) |

The 'TLS Connection Security' section is also visible below the Scan Configuration section.

| Field                                  | Value   |
|--|---|
| Trusted roots:                         | -- Please import Trusted Root for ICAP to CertStore --  |
| TLS options:                           | <input type="checkbox"/> Accept expired TLS certificates<br><input type="checkbox"/> Allow partial certificate chains |
| Certificate subject:                   |   |
| Certificate expiration warning period: | 21  |

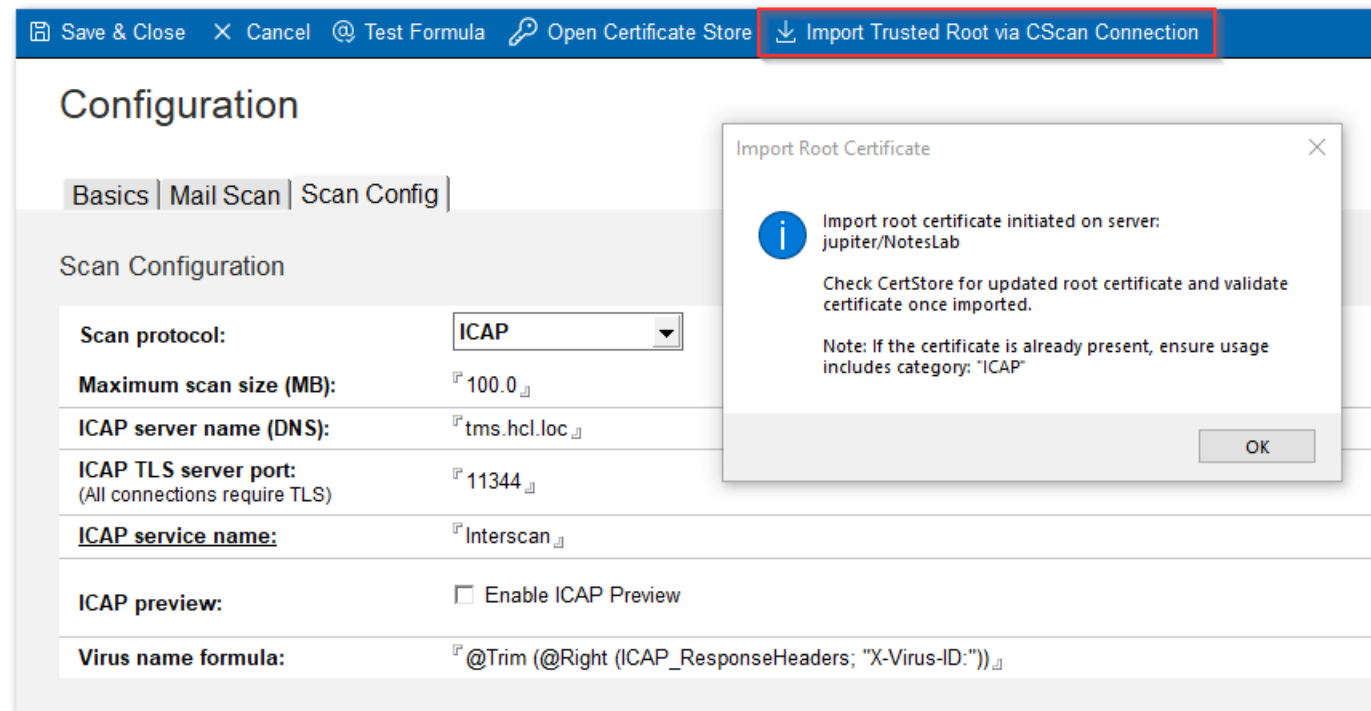
# Specify Scan Configuration / TLS Certificate

- TLS/SSL is required for all ICAP connections
  - A trusted root needs to be imported and assigned to the ICAP category
  - Trusted roots are **imported** into **certstore.nsf**
  - **CertMgr** and **certstore.nsf** are required for configuring for ICAP TLS connections!
  - In case no domain wide **certstore.nsf** has been created, refer to Domino 12
  - CertMgr runs on one server in the domain acting as management server for all certificate operations.
  - The server running ICAP requires a **certstore.nsf** replica
  - Tip: **Load certmgr** on any server will pull a **certstore.nsf** replica from CertMgr server

The screenshot shows a configuration window with a blue header bar containing buttons: Save & Close, Cancel, Test Formula, Open Certificate Store, and Import Trusted Root via CScan Connection. The main title is 'Configuration'. Below it are tabs: Basics, Mail Scan, and Scan Config. The 'Scan Configuration' section includes fields for: Scan protocol (ICAP), Maximum scan size (MB) (100.0), ICAP server name (DNS) (tms.hcl.loc), ICAP TLS server port (11344), ICAP service name (Interscan), ICAP preview (checkbox), and Virus name formula (@Trim (@Right (ICAP\_ResponseHeaders; "X-Virus-ID:"))). The 'TLS Connection Security' section includes: Trusted roots (a dropdown menu with a red box around it containing the text '-- Please import Trusted Root for ICAP to CertStore --'), TLS options (checkboxes for 'Accept expired TLS certificates' and 'Allow partial certificate chains'), Certificate subject, and Certificate expiration warning period (21).

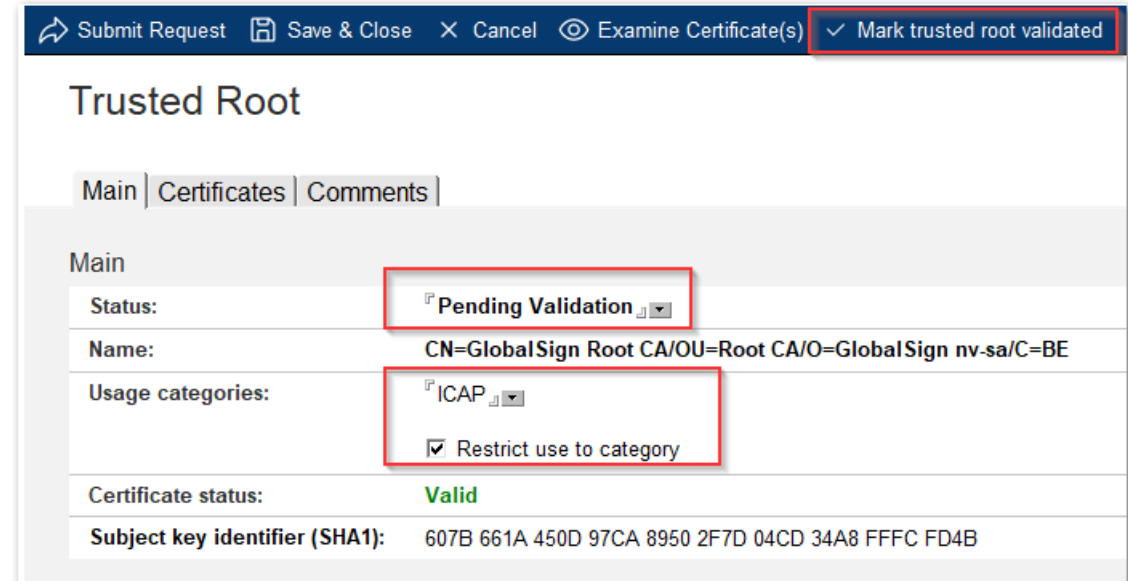
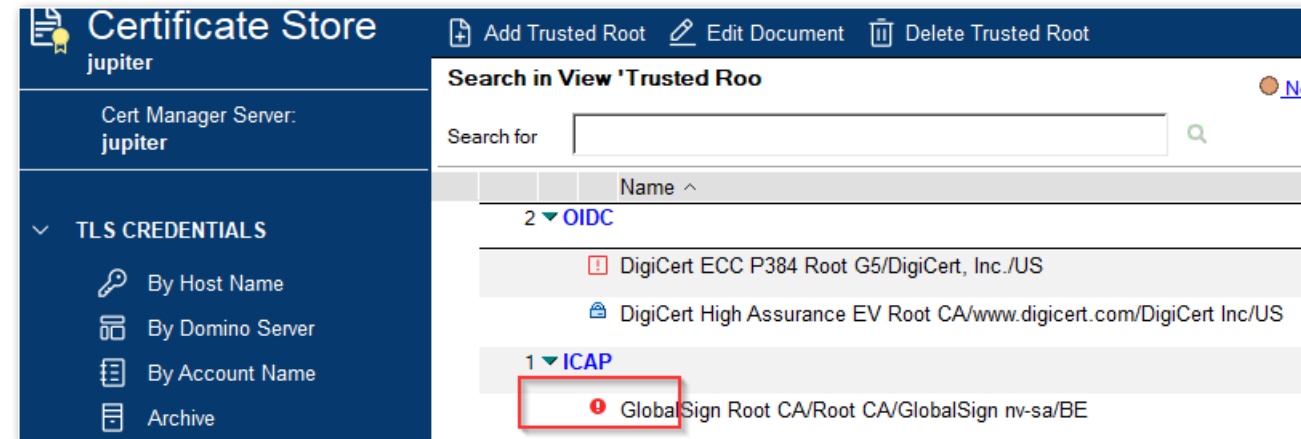
# Specify Scan Configuration / TLS Certificate

- Trusted Root import into **certstore.nsf** can be performed in multiple ways, based on the configuration
- If using a proper certificate with a **SAN** certificate, the **import wizard** can help to import the trusted root
  - The wizard will try to connect to the ICAP server to retrieve the trusted root
  - In case the trusted root is **not** send with the certificate chain, a lookup in Domino directory is performed to obtain the trusted root
- If the operation completed successfully, a new draft Trusted Root document is created in **certstore.nsf**



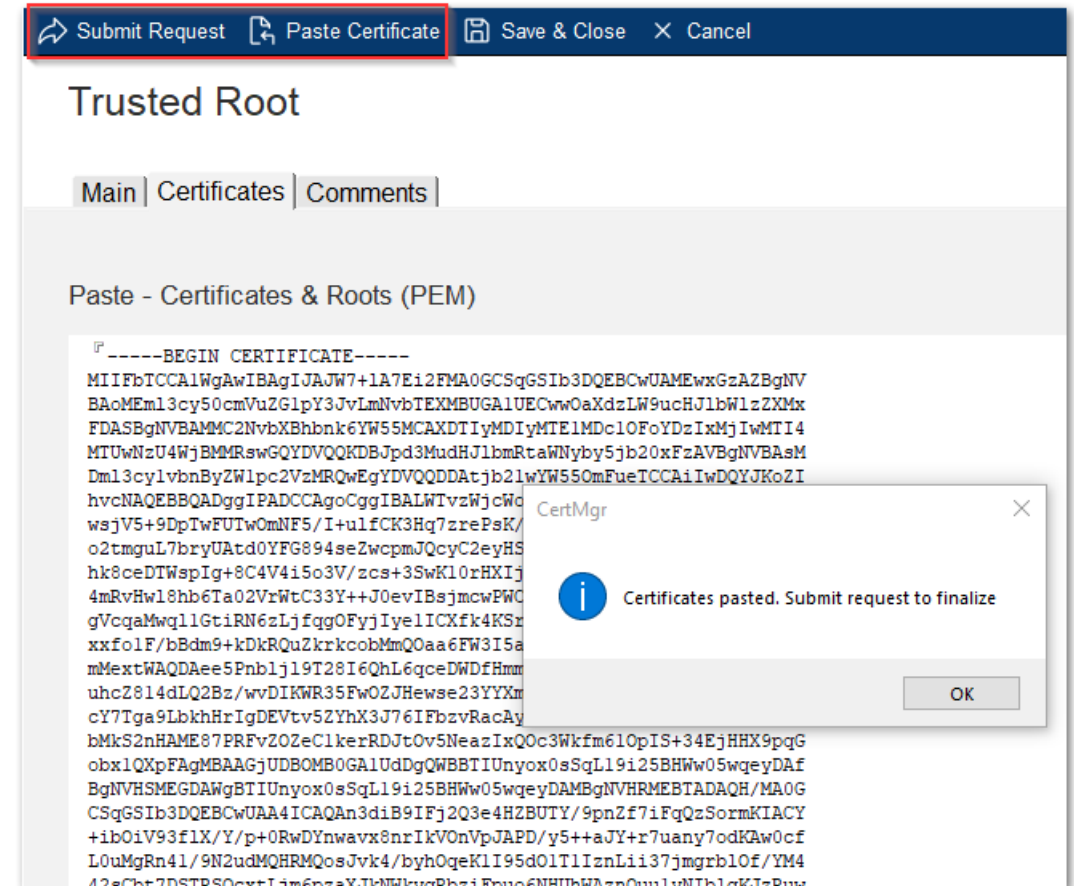
# Trusted Root for ICAP Connection

- Open **certstore.nsf** via action button from CScan configuration document
- If the wizard retrieved a trusted root, it will be marked for “**pending validation**”
  - The selected category is **ICAP**
  - If the certificate is already there, the category **ICAP** is only added to the trusted root document
  - The wizard always imports trusted roots in “**Pending Validation**” status and restricts the use to ICAP category
- If the trusted root should be also used for other use cases, remove “**Restrict use to category**”



# Manually import Trusted Root for ICAP Connection

- Some ICAP appliances ship with their own **self signed CA** without **SAN** (Subject Alternate Name) certificates
- Without a **SAN** the wizard cannot validate the certificate
- The trusted root can also be manually imported
  1. Create a new Trusted Root document
  2. Paste the PEM data
  3. Submit the request to **CertMgr** for processing
  4. Add the **ICAP** category to the newly created trusted root



# Resulting Trusted Root Certificate

- Check the resulting certificate
- Add the **ICAP** category to the newly created trusted root
- Some appliances use their own simple CA or self signed certificate
- In this case you might want to use **“Restrict use to category”**
- If the trusted root should be used for other use cases do not specify **“Restrict use to category”**

| Trusted Root                   |  |
|--------------------------------|--|
| Main   Certificates   Comments |  |
| Main                           |  |
| Status:                        | Issued   |
| Name:                          | CN=GlobalSign Root CA/OU=Root CA/O=GlobalSign nv-sa/C=BE             |
| Usage categories:              | ICAP<br><input checked="" type="checkbox"/> Restrict use to category |
| Certificate status:            | Valid  |
| Subject key identifier (SHA1): | 607B 661A 450D 97CA 8950 2F7D 04CD 34A8 FFFC FD4B                    |

# Verify CScan Trusted Root Configuration

- Return to **cscancfg.nsf** and refresh the document
- Verify the newly added trusted root is displayed
- By default all trusted roots in the **ICAP** category will be used
  - The trusted roots can be restricted to an explicit list with the selection option below the display field
- Some ICAP appliances cannot handle certificate chains with intermediate certs.
  - In this case select “**Allow partial certificate chains**” option and import the intermediate certificate
- Tip: The MicroCA can create an internal certificate valid for two years as well

The screenshot shows the 'Configuration' window with the 'Scan Config' tab selected. The 'Scan Configuration' section includes fields for 'Scan protocol' (set to ICAP), 'Maximum scan size (MB)' (100.0), 'ICAP server name (DNS)' (tms.hcl.loc), 'ICAP TLS server port' (11344), 'ICAP service name' (Interscan), 'ICAP preview' (unchecked), and 'Virus name formula' (@Trim (@Right (ICAP\_ResponseHeaders; "X-Virus-ID:"))). The 'TLS Connection Security' section shows 'Trusted roots' with a dropdown menu displaying 'GlobalSign Root CA/Root CA/GlobalSign nv-sa/BE'. Below this, 'TLS options' are shown with 'Accept expired TLS certificates' (unchecked) and 'Allow partial certificate chains' (checked, highlighted with a red box). The 'Certificate subject' field is empty.

| Configuration  |  |
|--|--|
| Basics   Mail Scan   Scan Config                       |  |
| Scan Configuration                                     |  |
| Scan protocol:   | ICAP   |
| Maximum scan size (MB):                                | 100.0  |
| ICAP server name (DNS):                                | tms.hcl.loc  |
| ICAP TLS server port:<br>(All connections require TLS) | 11344  |
| ICAP service name:                                     | Interscan  |
| ICAP preview:  | <input type="checkbox"/> Enable ICAP Preview   |
| Virus name formula:                                    | @Trim (@Right (ICAP_ResponseHeaders; "X-Virus-ID:"))   |
| TLS Connection Security                                |  |
| Trusted roots:   | GlobalSign Root CA/Root CA/GlobalSign nv-sa/BE   |
| TLS options:   | <input type="checkbox"/> Accept expired TLS certificates<br><input checked="" type="checkbox"/> Allow partial certificate chains |
| Certificate subject:                                   |  |



# Certificates without SAN

- **SAN** certificates are required by most applications today
  - But many ICAP appliances ship with simple self signed certificates out of the box
  - Many customers might still use those certificates
  - It is not recommended but commonly used
- CScan can **alternatively** verify the subject of the certificate in this case
- Specify the **exact subject** in the ICAP configuration
- In case the subject is wrong, the admin finds an error message including the expected subject name in the log

The screenshot shows the 'Configuration' window of an ICAP appliance. The 'Scan Config' tab is selected. The 'Scan Configuration' section includes fields for 'Scan protocol' (set to ICAP), 'Maximum scan size (MB)' (10.0), 'ICAP server name (DNS)' (tms.hcl.loc), 'ICAP TLS server port' (11344), 'ICAP service name' (Interscan), 'ICAP preview' (checked), and 'Virus name formula' (@Trim (@Right (ICAP\_ResponseHeaders; "X-Virus-ID:"))). The 'TLS Connection Security' section includes 'Trusted roots' (a list of certificates), 'TLS options' (checkboxes for 'Accept expired TLS certificates' and 'Allow partial certificate chains'), 'Certificate subject' (CN=company:any/OU=iws-onpremises/O=iws.trendmicro.com, highlighted with a red box), and 'Certificate expiration warning period' (21).

| Configuration  |   |
|--|---|
| Basics   Mail Scan   Scan Config                       |   |
| Scan Configuration                                     |   |
| Scan protocol:   | ICAP  |
| Maximum scan size (MB):                                | 10.0  |
| ICAP server name (DNS):                                | tms.hcl.loc   |
| ICAP TLS server port:<br>(All connections require TLS) | 11344   |
| ICAP service name:                                     | Interscan   |
| ICAP preview:  | <input checked="" type="checkbox"/> Enable ICAP Preview   |
| Virus name formula:                                    | @Trim (@Right (ICAP_ResponseHeaders; "X-Virus-ID:"))  |
| TLS Connection Security                                |   |
| Trusted roots:   | company:any/iws-onpremises/iws.trendmicro.com<br>GlobalSign Root CA/Root CA/GlobalSign nv-sa/BE                       |
| TLS options:   | <input type="checkbox"/> Accept expired TLS certificates<br><input type="checkbox"/> Allow partial certificate chains |
| Certificate subject:                                   | CN=company:any/OU=iws-onpremises/O=iws.trendmicro.com   |
| Certificate expiration warning period:                 | 21  |



# Create Server Configuration

- Create a new server configuration for a server
- Select the server name
- Select the configuration just created
  - If only one configuration is present, the configuration is automatically selected
- Each server can only have one configuration
  - The selection dialog hides servers with existing config
  - Note: Configurations can only be deleted if no server is assigned
- Once configured the Health Check status of the configurations is “**Pending validation**”
  - **mailscan** server task will validate the connection and set the health check status

Save & Close × Cancel Open Configuration Open Certificate Store × Disable Virus Scan

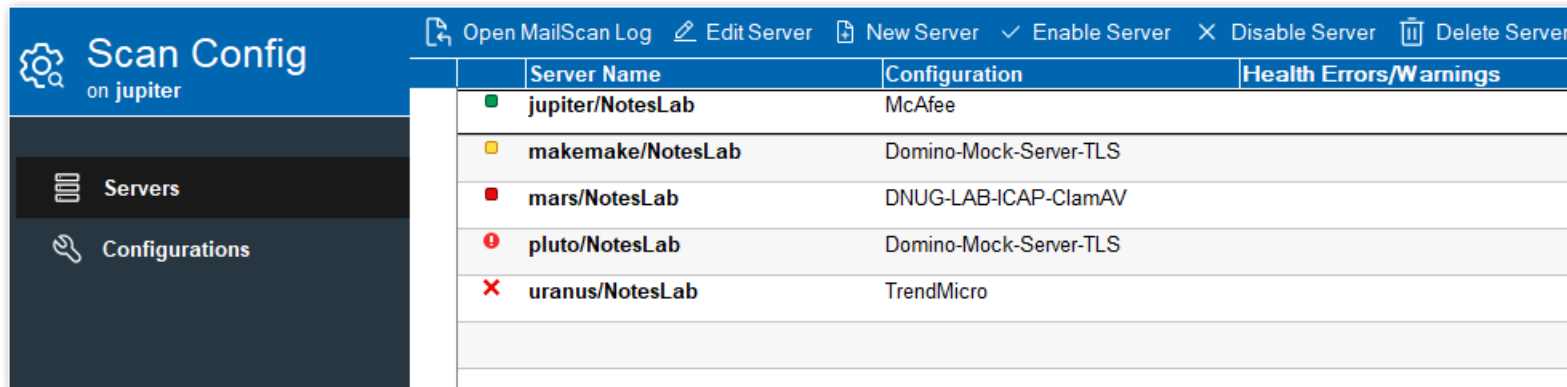
### Server

Main

|                     |                                      |
|---------------------|--------------------------------------|
| Server name:        | jupiter/NotesLab                     |
| Configuration name: | TrendMicro                           |
| Health status:      | Pending validation                   |
| Status:             | Enabled                              |
| Log options:        | normal                               |
|                     | <input type="checkbox"/> Log to file |
| Comments:           |                                      |

# CScan Server Status

- Server status view shows the status per server
- Includes important server errors directly written into server document by **mailscan** servertask
- Action to open the log database directly from the view



| Server Name       | Configuration          | Health Errors/Warnings |
|-------------------|------------------------|------------------------|
| jupiter/NotesLab  | McAfee                 |                        |
| makemake/NotesLab | Domino-Mock-Server-TLS |                        |
| mars/NotesLab     | DNUG-LAB-ICAP-ClamAV   |                        |
| pluto/NotesLab    | Domino-Mock-Server-TLS |                        |
| uranus/NotesLab   | TrendMicro             |                        |

# CScan Log Database

- Log per attachment
  - Only shows viruses unless in test mode to log all attachments
- Log per message shows sender/recipient and all logged attachments

| Scan Log<br>on jupiter |                        | Scanned | FileName                                 | MB/sec | Size MB | Sec  | SHA1                                     | Comp |     |
|------------------------|------------------------|---------|--|--------|---------|------|--|------|-----|
| 4 ▼ OK                 |                        |         |  |        |         |      |  |      |     |
| 4 ▼ jupiter/NotesLab   |                        |         |  |        |         |      |  |      |     |
| 3 ▼ mail1.box          |                        |         |  |        |         |      |  |      |     |
|                        | 10/06/2022 04:09:50 PM | ✓       | 51871791_01-Aug-2022-31-Aug-2022_360813: | 0.1    | 0.0     | 0.3  | 05AADD19A294A5AAF88F6CD07D9E71D164AC6DBF | LZ1  | 204 |
|                        | 09/07/2022 10:51:17 PM | ✓       | eicar.com.txt                            | 0.0    | 0.0     | 0.0  | 1500490B84329370384D83217FA59BCAE6F0507E |      | 204 |
|                        | 09/07/2022 10:47:40 PM | ✓       | eicar.com.txt                            | 0.0    | 0.0     | 0.1  | 1500490B84329370384D83217FA59BCAE6F0507E |      | 204 |
| 1 ▼ mail3.box          |                        |         |  |        |         |      |  |      |     |
|                        | 10/04/2022 11:56:01 PM | ✓       | HCL_Notes_12.0.1FP1_Win.exe              | 2.0    | 160.5   | 82.2 | 17F07E6B8DB243177C58C774AF34A9A69F43B955 | LZ1  | 204 |
| 4 ▼ Virus              |                        |         |  |        |         |      |  |      |     |
| 4 ▼ jupiter/NotesLab   |                        |         |  |        |         |      |  |      |     |
| 4 ▼ mail1.box          |                        |         |  |        |         |      |  |      |     |
|                        | 09/23/2022 04:09:03 PM | ✗       | eicar.com.txt                            | 0.0    | 0.0     | 0.3  | 3395856CE81F2B7382DEE72602F798B642F14140 |      | 403 |
|                        | 09/22/2022 11:50:37 PM | ✗       | eicar.com.txt                            | 0.0    | 0.0     | 0.3  | 3395856CE81F2B7382DEE72602F798B642F14140 |      | 403 |
|                        | 09/13/2022 05:43:08 PM | ✗       | eicar.com.txt                            | 0.0    | 0.0     | 0.0  | 3395856CE81F2B7382DEE72602F798B642F14140 |      | 403 |
|                        | 09/07/2022 10:44:19 PM | ✗       | eicar.com.txt                            | 0.0    | 0.0     | 0.0  | 3395856CE81F2B7382DEE72602F798B642F14140 |      | 403 |
| 8                      |                        |         |  |        |         |      |  |      |     |

# CScan Log for Attachments

- Contains details for each attachment
- Details about virus found & status returned by ICAP
- Lookup for SHA1 hash on **Virus Total website**

| VirusTotal Lookup Open Message Log |  |
|------------------------------------|--|
| Attachment Log                     |  |
| Main   ICAP                        |  |
| Status:                            | Virus                                    |
| Virus Information:                 | Eicar_test_file                          |
| Scan Time:                         | 10/08/2022 10:26:05 AM CEDT              |
| Server name:                       | jupiter/NotesLab                         |
| Database:                          | mail1.box                                |
|                                    |  |
| File name:                         | eicar.com.txt                            |
| Runtime (seconds):                 | 0.39                                     |
| File Size (MB):                    | 0.00                                     |
| File hash (SHA1):                  | 3395856CE81F2B7382DEE72602F798B642F14140 |
| Note UNID:                         | 2CF803BFCB0E58D1C12588D5002E54F3         |
| Replica ID:                        | C125880E:0020A470                        |

| VirusTotal Lookup Open Message Log |  |
|------------------------------------|--|
| Attachment Log                     |  |
| Main   ICAP                        |  |
| ICAP Status:                       | ICAP/1.0 200 OK  |
| HTTP Status:                       | HTTP/1.1 403 Forbidden (TMWS Blocked)  |
|                                    |  |
| Response ICAP Headers:             | Connection: keep-alive<br>Date: Sat, 08 Oct, 2022 08:26:05 GMT<br>Encapsulated: res-hdr=0, res-body=180<br>Server: TMWS<br>ISTag: "TMWS 5 5638"<br>X-Virus-ID: Eicar_test_file<br>X-Infection-Found: Type=0; Resolution=2; Threat=Eicar_test_file; |
| Response HTTP Headers:             | Date: Sat, 08 Oct 2022 08:26:05 GMT<br>Cache-Control: no-cache<br>Content-Type: text/html; charset=UTF-8<br>Server: TMWS<br>Content-Length: 49825  |
|                                    |  |
| Service ID:                        | TrendMicro   |

|                        |   |
|------------------------|---|
| ICAP Status:           | ICAP/1.0 200 OK   |
| HTTP Status:           | HTTP/1.1 403 VirusFound   |
|                        |   |
| Response ICAP Headers: | ISTag: "007849-25.156-010493-111791"<br>Encapsulated: res-hdr=0, res-body=122                       |
| Response HTTP Headers: | Content-Type: text/html<br>Cache-Control: no-cache<br>Content-Length: 2703<br>X-Frame-Options: deny |
|                        |   |
| Service ID:            |   |
| Service :              | McAfee Web Gateway 10.2.9 build 40478   |

# CScan Log per message

- Embedded view for attachments
- Quarantine link only shown, if quarantine document is available
- Attachment log, message log and quarantine document are linked via **ReplicaID/UNID** fields

The screenshot displays the 'Message Log' interface. At the top, there is a blue bar with a button labeled 'Open Quarantine Message'. Below this, the 'Message Log' title is followed by tabs for 'Main' and 'Details'. The 'Main' tab is active, showing 'Message Information'.

**Message Information**

|                |   |
|----------------|---|
| Posted date:   | 05/13/2022 04:59 PM                                     |
| Subject:       | cscan: virus found & message blocked - Eicar Test Virus |
| From:          | Daniel Nashed/Germany                                   |
| Sender:        | Daniel Nashed/Germany                                   |
| Recipients:    | nsh@nashcom.de  |
| Sent to:       | nsh@nashcom.de  |
| Copy to:       |   |
| Blind copy to: |   |

**Virus Information**

**Attachments**

| Scanned                | FileName        | MB/sec | Size MB | Sec | SHA1                                 |
|------------------------|-----------------|--------|---------|-----|--------------------------------------|
| 10/08/2022 10:51:34 AM | ✘ eicar.com.txt | 0.0    | 0.0     | 0.1 | 3395856CE81F2B7382DEE72602F798B642F1 |

A warning dialog box is overlaid on the message details, titled 'Open infected quarantine document?'. It contains a yellow warning icon and the text: 'You are opening a document, with infected attachments! Are you sure you want to open the document?'. The dialog has 'Yes' and 'No' buttons.

# Scan Status Token “\$\$CScanToken”

- Each server creates a modern **Ed25519** private key (<https://en.wikipedia.org/wiki/Curve25519> )
  - Private key is encrypted for server and stored in **cscancfg.nsf** server configuration document
  - Used to sign a JWT scan status token
- Public key is also stored in server configuration document to allow other servers to verify the token
  - Each server uses a public key cache for the validation of other server's tokens

### Example of Scan Token

Field Name: \$\$CScanToken

```
"eyJ0eXAiOiAiSldUliwglmFsZyl6lCJFZERTQsJ9.eyJ2ZX.  
MjlxMDA4VDE5NTgyMyw5MiswMCIsbnNlcnZlcil6lkNOPV  
yLTAXMDQ5NC0xMTE4MDJcllslmNvbWZpZ0RiljoiQzEyNTg0OTQxMDY4NZA4MlslmNvbWZpZ0RlEjoiNzIiCQKQ2MzgxOTFGOEQyOEMxM  
jU4ODVBMDA2OUlxQkliLCJjb25maWdOYW1lIjoiTWNBZmVlliwidmVyaWZpY2F0aW9uSGFzaCI6IkRBMzIiBM0VFNUU2QjRCMEQzMjU1Q  
kZFRjk1NjAxODkwQUZEODA3MDkiLCJrZXIuUaHVtYnByaW50IjojMXg1ZjhnWFNSSnQ1OVJYQUYyaig0TjJ4b1FVliwiaGFzaEFsZ29yaXRob  
Sl6llNIQTEifQ.B 8bB-nmkr9 bTaCZBkgz Px7Zpd7xLmlXsFC8 6ZaJ5y1dTqO0k4akoZ8jyuQ3nqVVB5l2oQLr0KFjmsKO3BA"
```

### Example of a public key

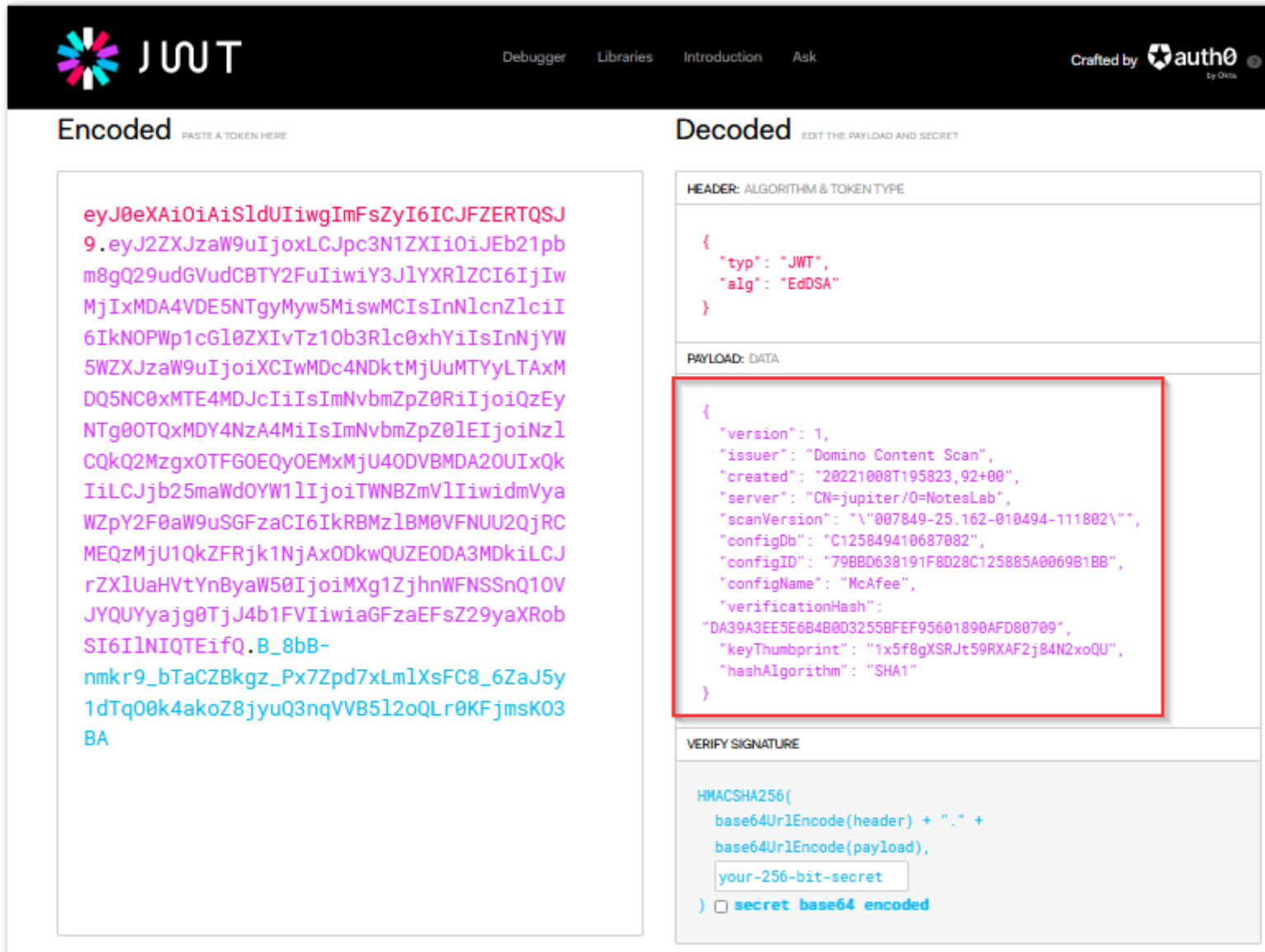
Field Name: PublicKey

"-----BEGIN PUBLIC KEY-----

MCowBQYDK2VwAyEAR+BSCPgf5lYhiLShYpgJBuaaYnU0qu53Qy4EWqmvo8k=

-----END PUBLIC KEY-----"

# Decode \$\$CScanToken



The screenshot shows the JWT.io web application. The 'Encoded' tab on the left contains a long base64-encoded string. The 'Decoded' tab on the right shows the token's structure:

- HEADER: ALGORITHM & TOKEN TYPE**

```
{  "typ": "JWT",  "alg": "EdDSA"}
```
- PAYLOAD: DATA**

```
{  "version": 1,  "issuer": "Domino Content Scan",  "created": "20221008T195823,92+00",  "server": "CN=jupiter/0=NotesLab",  "scanVersion": "\"007849-25.162-010494-111802\"",  "configDb": "C125849410687082",  "configID": "798BD638191F8D28C125885A0069B1BB",  "configName": "McAfee",  "verificationHash": "DA39A3EE5E6B4B0D3255BFEF95601890AFD80709",  "keyThumbprint": "1x5f8gXSRJt59RXAF2j84N2xoQU",  "hashAlgorithm": "SHA1"}
```
- VERIFY SIGNATURE**

```
HMACSHA256(  base64UrlEncode(header) + "." +  base64UrlEncode(payload),  your-256-bit-secret) ☐ secret base64 encoded
```

- JWT token can be decoded
- Details about JWT and a decoder can be found here -> <https://jwt.io>

# “\$\$CScanToken” decoded

- Payload contains information about
  - Virus scanner version
  - Configuration
  - Scan date
  - Reference to the key used
  - Verification hash of attachments

```
{
  "version": 1,
  "issuer": "Domino Content Scan",
  "created": "20221008T195823,92+00",
  "server": "CN=jupiter/O=NotesLab",
  "scanVersion": "\"007849-25.162-010494-111802\"",
  "configDb": "C125849410687082",
  "configID": "79BBD638191F8D28C125885A0069B1BB",
  "configName": "McAfee",
  "verificationHash": "DA39A3EE5E6B4B0D3255BFEF95601890AFD80709",
  "keyThumbprint": "1x5f8gXSRJt59RXAF2j84N2xoQU",
  "hashAlgorithm": "SHA1"
}
```



## Secure Domino Backup

- Backup approaches & tips
- New Windows VSS Writer Support



# Domino Backup

- **Backup & disaster recovery** should be part of your Domino security concept
- Ensure your backup strategy protects you against **ransomware attacks**, too!
- Backup repositories should not be writable at run-time
- If you use the basic Domino 12 Backup functionality to file storage your Domino server and the OS has access to **all backup files** – Not just the current backup!
- There is no one size fits all approach
- Depends on your backup integration and your environment

# Secure Backup Approaches

- Only mount/unmount volume in pre/post backup/restore operations
  - Does only make it less likely! → Still vulnerable during backup/restore!
- **Take a backup or snapshot** of the target storage and/or make files read-only after backup
  - For example if the backend storage is ZFS
  - ZFS can also encrypt file-systems and send snapshots to remote locations (without the key!)
- Use a backup solution with a secure repository
  - e.g. **Borg Backup** on Linux with remote repositories secured by **SSH/SFTP** (<https://www.borgbackup.org/> )
  - Free Restic backup in combination with **VSS Snapshots** and **SSH/SFTP** or **REST server** (<https://restic.net/>)
- Use a commercial solution like **Veeam Backup & Replication** to protect your backups

# Domino VSS Writer?

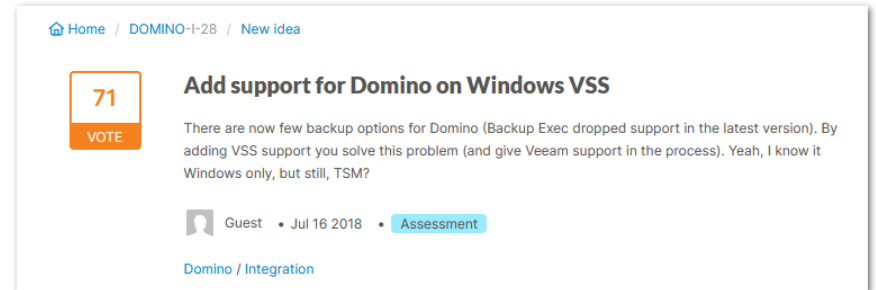
- VSS Admin Windows tool shows all registered VSS Writers
  - VSS Write support does not require any backup integration
- The AHA idea was also high on my personal wish list
- There was not Domino VSS Writer support ... until now

```
vssadmin list writers
```

```
vssadmin 1.1 - Volume Shadow Copy Service administrative command-line tool  
(C) Copyright 2001-2013 Microsoft Corp.
```

```
Writer name: 'Domino Backup Writer'  
  Writer Id: {b95d0c5e-57d4-412b-b571-18a81a16abba}  
  Writer Instance Id: {287e5f15-b760-4024-9719-4b995206faf5}  
  State: [1] Stable  
  Last error: No error
```

```
Writer name: 'Registry Writer'  
  Writer Id: {afbab4a2-367d-4d15-a586-71dbb18f8485}  
  Writer Instance Id: {627f7844-3a6c-4202-a4d6-1886edbf5c06}  
  State: [1] Stable  
  Last error: No error
```



Home / DOMINO-I-28 / New idea

**71**  
VOTE

**Add support for Domino on Windows VSS**

There are now few backup options for Domino (Backup Exec dropped support in the latest version). By adding VSS support you solve this problem (and give Veeam support in the process). Yeah, I know it Windows only, but still, TSM?

Guest • Jul 16 2018 • Assessment

[Domino / Integration](#)

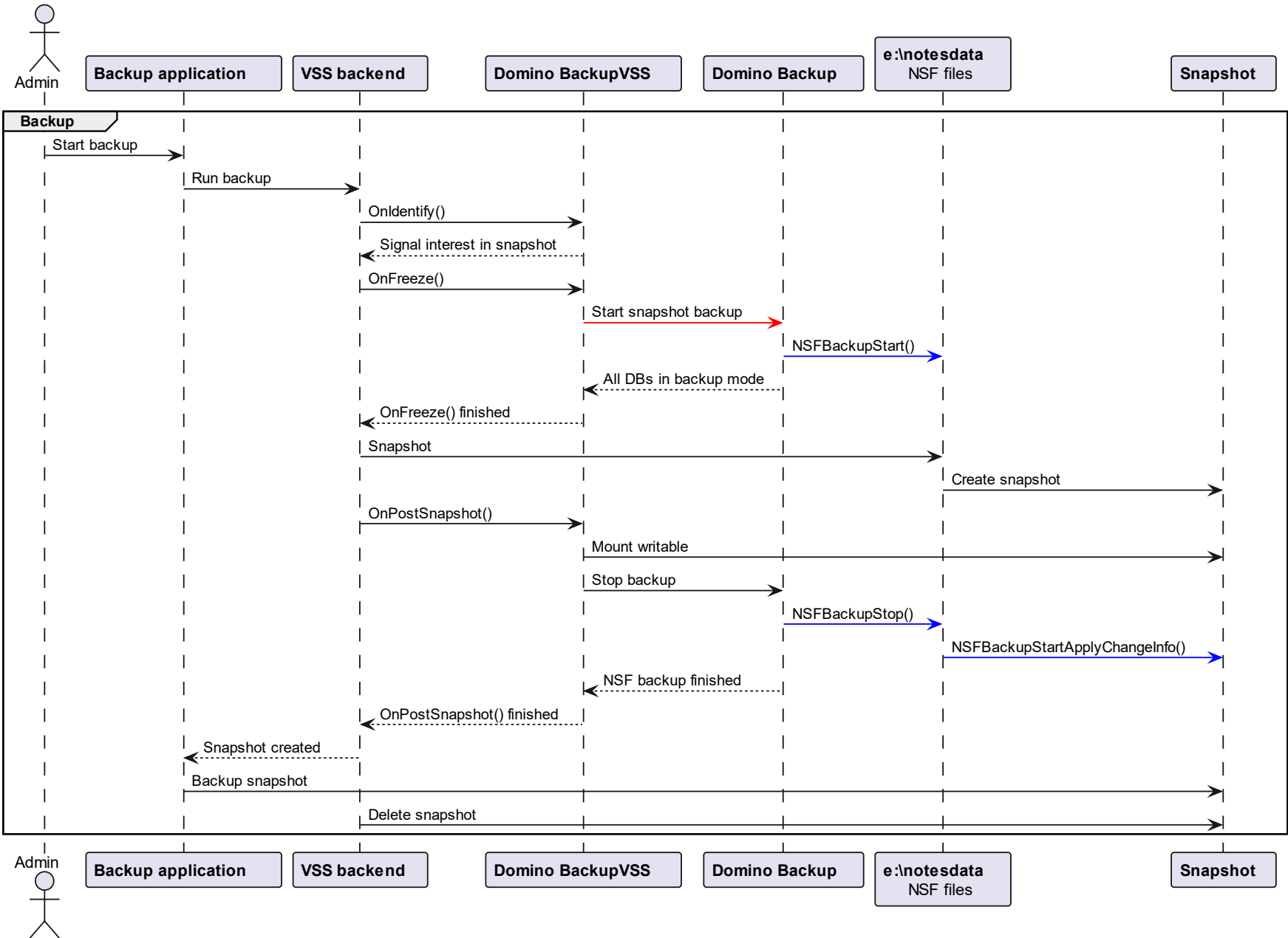
# Windows VSS Writer Support

- Volume Shadow Copy Service (VSS) supports application integrated **snapshot operations**
- “**VSS Writer**” allows to make an application fully “**snapshot aware**” without any direct backup application integration or scripting
- Requires Domino to become a “**VSS Writer**”
- **Flow**
  - Domino registers as a “**VSS Writer**” using a **Microsoft VSS API**
  - **Backup application** starts **VSS Snapshot**
  - **Windows VSS** sends event to all **VSS Writers** registered to “**Freeze**” their application
  - Windows takes **VSS snapshot**
  - Windows sends VSS “**Post Thaw**” event to application
  - Domino processes delta data accumulated during snapshot operations

# VSS Writer “AutoRecover” Support

- The biggest challenge in the snapshot backup world
  - Snapshots cannot be modified
  - **Delta changes** need to be stored separately and need to be applied to the database on restore to make the NSF file consistent
- **Solution**
  - VSS Writers **VSS\_VOLSNAP\_ATTR\_AUTORECOVER** Option
  - Allows a **VSS Writer** to update the snapshot in the **OnPostSnapshot** event to
    - Merge **delta** information occurred during backup
    - Make the database consistent for recovery without Domino restore operations
  - The Domino VSS Writer supports **AutoRecovery** to apply changes directly into the writable snapshot in **OnPostSnapshot** event

# Domino Backup VSS Writer Flow



# Implementation

- Separate “**backupvss**” server task registering as a **VSS backup** writer
- Invokes Domino “**backup**” server task to leverage “Domino Backup Snapshot Mode”
- In **Freeze** event waits for backup task to be in snapshot before signaling the snapshot can be created
- Integrates **VSS Writer functionality** into Core Domino
  - Separate task is needed to control “**backup**” server task
  - “**backupvss**” task is required to be permanently loaded to allow **VSS backend** to communicate with Domino
- **GitHub**: Updated, simplified Domino 12.0.2 Veeam integration for restore only
  - [https://opensource.hcltechsw.com/domino-backup/backup-providers/veeam/install\\_vss\\_writer](https://opensource.hcltechsw.com/domino-backup/backup-providers/veeam/install_vss_writer)



# VSS Writer Implementation Limitations

- **NSF Data is required to be on a single volume for snapshot**
  - No support for external directory or NSF links pointing to a different volume
  - No support for Windows junctions and comparable mount options
  - Support for directory and NSF link on the same physical volume
- VSS Snapshot backup application requires to support “**AutoRecovery**” mode for full functionality
  - Fallback to write delta files is still possible – In the same way it is supported in 12.0.1 today
- **Restore still requires separate integration similar to Veeam integration available today**
  - Restore integration scripts are posted in **GitHub repository**
  - No support for VSS restore operations
  - Vendors backup to their own repository and have no direct VSS restore integration
- Only one Domino partition per Windows machine can be backed up via VSS

# HCL Documentation & Projects

- **Domino 12.0.2 New security features and enhancements**
  - [https://help.hcltechsw.com/domino/12.0.2/admin/wn\\_security.html](https://help.hcltechsw.com/domino/12.0.2/admin/wn_security.html)
  - [https://help.hcltechsw.com/domino/12.0.2/admin/wn\\_security1201.html](https://help.hcltechsw.com/domino/12.0.2/admin/wn_security1201.html)
- **HCL GitHub – CertMgr**
  - <https://github.com/HCL-TECH-SOFTWARE/domino-cert-manager>
- **HCL GitHub – Domino Backup**
  - <https://opensource.hcltechsw.com/domino-backup/>
- **HCL GitHub – Domino Container Community Project**
  - <https://opensource.hcltechsw.com/domino-container/>

# Further Reading

- **GitHub – Domino Start Script Project**
  - <https://nashcom.github.io/domino-startscript/>
- **Blog Post: Domino V12 using CertMgr for certificates used outside Domino**
  - <https://blog.nashcom.de/nashcomblog.nsf/dx/domino-v12-using-certmgr-for-certificates-used-outside-domino.htm>
- **Blog Post: NGINX CertMgr Integrations**
  - <https://blog.nashcom.de/nashcomblog.nsf/dx/using-domino-certmgr-with-nginx-co.htm>
- **Blog Post: Leveraging Domino Event Monitoring for Domino V12 CertMgr Health Checks**
  - <https://blog.nashcom.de/nashcomblog.nsf/dx/leveraging-domino-event-monitoring-for-domino-v12-certmgr-health-checks.htm>
- **Blog Post: Fail2Ban Support for Domino on Linux -- Intrusion Detection**
  - <https://blog.nashcom.de/nashcomblog.nsf/dx/fail2ban-support-for-domino-intrusion-detection.htm>

# Questions & Answers



- **Thank you** for your interest in “Domino 12.0.1 + 12.0.2 Security”
- **Open questions in chat?**
  - Presentation will be available for download from OpenNTF
  - There will be a Q&A summary on OpenNTF
- Additional information
  - <https://blog.nashcom.de>
  - [nsh@nashcom.de](mailto:nsh@nashcom.de)

