Dokumentation Abschlussprojekt

GEARFISH TEAM 12

Keanu Amann · David Kleinfercher

Version	1.3
Stand	27. Mai 2026
Domain	t12.lan
Klassifizierung	intern
Begleitdokumente	Veeam Dokumentation · Naemon Dokumentation Rede Server / Client Dokumentation · Projektauftrag · Lastenheft · Pflichtenheft

ÜBER DIESES DOKUMENT

Dieses Dokument fasst die gesamte Infrastruktur des Abschlussprojekts zusammen: Netzwerklayout und WAN-Anbindung, Proxmox-Hosts, FortiGate-Firewall inklusive SSL-VPN und Policy-Set, Active Directory, alle Server im LAN und in der DMZ sowie die wichtigsten Betriebs- und Wartungsthemen (Backup, Monitoring, Autounattend, Baramundi-Lizenz, Hygiene-TODOs, bekannte Probleme).

> **Credentials:** Alle Passwörter, Tokens und Secrets sind ausschließlich im **KeePass-Tresor** (team12_credentials.kdbx) hinterlegt. In diesem Dokument stehen nur Benutzernamen, Hosts, Ports und URLs.
>
> **Stand der Firewall-Sektion:** Sektion 3 (FortiGate) wurde auf die live ausgelesene Config (#config-version=FGT50E-6.2.12-FW-build1319-221102) abgeglichen. Hygiene-TODOs aus diesem Abgleich finden sich in §10z

INHALT

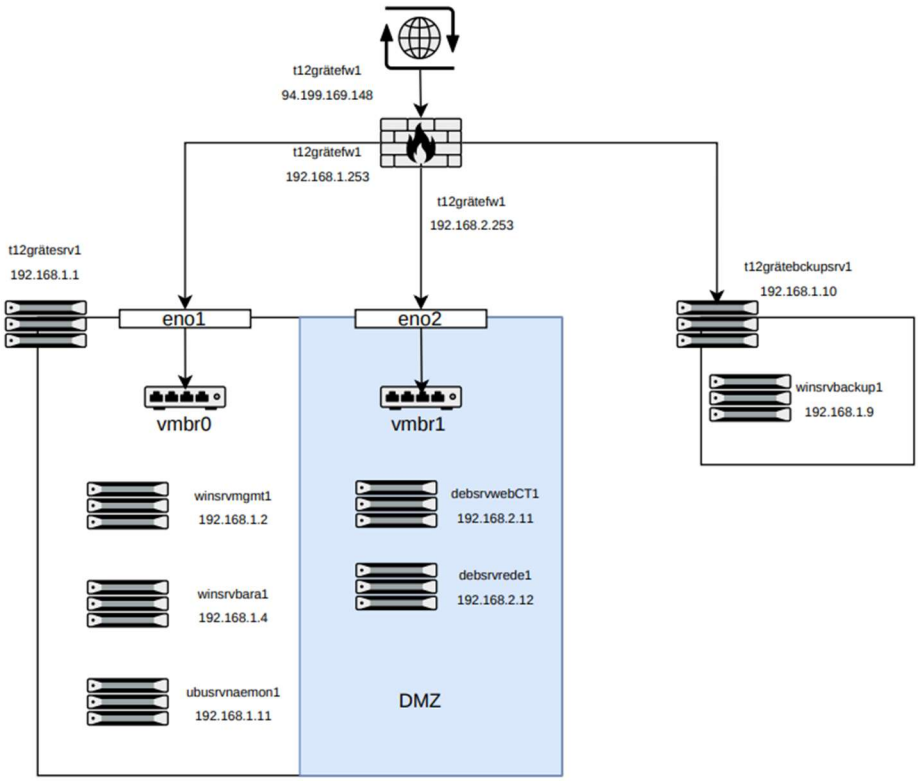
1. Allgemeine Informationen

1.1 NAMING-SCHEMA

```
[OS] + [Device-Type] + [Function] + [Device-Count]
```

Domain: t12.lan

1.3 NETZWERK-ÜBERSICHT



1.4 SUBNETZ-ÜBERSICHT

Segment	**Netz**	**Gateway**
LAN	192.168.1.0/24	192.168.1.253 (Firewall)
DMZ	192.168.2.0/24	192.168.2.253 (Firewall)
VPN-Tunnel-Pool	10.212.134.200 10.212.134.210	-
FortiLink (Switch-Mgmt)	169.254.1.0/24	169.254.1.1 (Firewall)

1.5 WAN-ANBINDUNG

Parameter	**Wert**
Öffentliche IPv4	94.199.169.148/28
Gateway	94.199.169.158
DNS primär	94.199.168.1
DNS sekundär	94.199.169.1

2. Proxmox-Server

Die gesamte Server-Landschaft des Projekts läuft virtualisiert auf zwei Proxmox-VE-Hosts. Proxmox dient dabei als Typ-1-Hypervisor: Statt für jeden Dienst eigene Hardware vorzuhalten, werden alle Server als virtuelle Maschinen bzw. Container auf diesen beiden Knoten betrieben.

Eine Besonderheit des Setups: Bei der „Hardware" handelt es sich nicht um echte Server, sondern um **Kemp-Loadbalancer-Appliances, die zweckentfremdet als Proxmox-Hosts** eingesetzt werden. Das bringt deutliche Hardware-Beschränkungen mit sich (begrenzte CPU-, RAM- und Storage-Ressourcen) und ist der Hauptgrund, warum im gesamten Projekt besonderes Augenmerk auf Monitoring (siehe naemon_dokumentation.md) und Backups (siehe veeam_dokumentation.md) liegt — die Hardware ist nicht für diesen Einsatzzweck ausgelegt und entsprechend ausfallgefährdet.

Die beiden Knoten sind klar aufgeteilt:

- **Gräte-Server (192.168.1.1)**: trägt die gesamte produktive Infrastruktur: alle LAN- und DMZ-VMs (AD/DNS/DHCP, Baramundi, Naemon, Web- und Rede-Server).
- **Backup-Server (192.168.1.10)**: beherbergt aufgrund der Ressourcenbeschränkungen ausschließlich die Backup-VM (winsrvbackup1 mit Veeam). Eine Lastverteilung der produktiven VMs auf beide Knoten ist mangels Kapazität nicht möglich.

Beide Hosts sind selbst Teil des Monitorings: Über den Serviceuser kairos und die installierten nagios-plugins werden sie von Naemon mitüberwacht.

2.1 GRÄTE-SERVER

Feld	Wert
IP	192.168.1.1
root	*Passwort im KeePass*
amke	*Passwort im KeePass*
kairos	*Passwort im KeePass* (Service-User, kein sudoer)

2.2 BACKUP-SERVER

Feld	Wert
IP	192.168.1.10
root	*Passwort im KeePass*
Admin	*Passwort im KeePass*
kairos	*Passwort im KeePass* (Service-User, kein sudoer)

2.3 GEMEINSAME PAKETE

Auf beiden Proxmox-Hosts installiert:

- nagios-plugins-basic (für Naemon-Queries)
- nagios-plugins-contrib (für Naemon-Queries)
- vim

3. Firewall (FortiGate)

3.1 GERÄTEINFORMATIONEN

Feld	**Wert**
Hostname	amke-klda-fortigate
Modell	FortiGate-50E
Firmware	v6.2.12, build 1319 (GA)
Seriennummer	FGT50E5619054806
Operation Mode	NAT
Geo-Position (Asset)	47.2531 N, 9.6122 E

3.2 ADMIN-ZUGANG

Feld	**Wert**
Benutzer	admin (Accprofile super_admin, VDOM root)
Passwort	*Passwort im KeePass*
Port	4443 (HTTPS)
Zugriff (LAN)	https://192.168.1.253:4443

3.3 GLOBALE EINSTELLUNGEN & DNS

```
config system global
    set admin-sport 4443
    set alias "FortiGate-50E"
    set hostname "amke-klda-fortigate"
    set switch-controller enable
    set timezone 26
end

config system dns
    set primary 94.199.168.1
    set secondary 94.199.169.1
end
```

3.4 INTERFACES

Name	Type	Members	IP/Netmask	Administrative Access
Hardware Switch 2				
fortilink	Hardware Switch		Dedicated to FortiSwitch	PING Security Fabric Connection
lan	Hardware Switch	lan1 lan3 lan4 lan5	192.168.1.253/255.255.255.0	PING HTTPS SSH HTTP +2
Physical Interface 3				
hierkommtsrein (wan1)	Physical Interface		94.199.169.148/255.255.255.240	PING
lan2	Physical Interface		192.168.2.253/255.255.255.0	PING HTTPS SSH
wan2	Physical Interface		0.0.0.0/0.0.0.0	PING FMG-Access

wan1 (WAN, aktiv)

```
edit "wan1"
    set ip 94.199.169.148 255.255.255.240
    set allowaccess ping https ssh
    set type physical
    set alias "hierkommtsrein"
    set role wan
next
```

- Erlaubte Dienste: ping, https, ssh
- Alias: hierkommtsrein

lan (LAN)

```
edit "lan"
    set ip 192.168.1.253 255.255.255.0
    set allowaccess ping https ssh http fgfm fabric
    set type hard-switch
    set stp enable
    set role lan
next
```

- Erlaubte Dienste: ping, https, ssh, http, fgfm, fabric
- Typ: hard-switch mit STP über die Ports lan1, lan3, lan4, lan5

lan2 (DMZ)

```
edit "lan2"
    set ip 192.168.2.253 255.255.255.0
    set allowaccess ping https ssh
    set type physical
    set role dmz
next
```

- Erlaubte Dienste: ping, https, ssh

ssl.root (VPN-Tunnel-Interface)

- Typ: tunnel
- Alias: SSL VPN interface
- Wird automatisch erstellt, sobald SSL VPN konfiguriert ist.

Weitere konfigurierte Interfaces (derzeit ungenutzt)

Interface	Modus	Status / Verwendung
wan2	DHCP, allowaccess ping fgfm	Backup-WAN-Port, aktuell kein Uplink
modem	PPPoE	Modem-Port, ungenutzt
fortilink	hard-switch, 169.254.1.1/24, eigener DHCP-Scope 169.254.1.2 169.254.1.254	Autoprovisioning-Interface für FortiSwitch/FortiExtender; aktuell sind keine angeschlossen

3.5 ROUTING

Default-Route über wan1 zum Gateway 94.199.169.158:

```
config router static
  edit 1
    set gateway 94.199.169.158
    set distance 5
    set device "wan1"
  next
end
```

3.6 FIREWALL-ADRESSEN

Adressobjekt	Inhalt
SSLVPN_TUNNEL_ADDR1	IP-Range für VPN-Clients: 10.212.134.200 10.212.134.210, gebunden an ssl.root
LAN_net	Subnetz 192.168.1.0/24 für das gesamte LAN
DMZ_net	Subnetz 192.168.2.0/24 für die DMZ

Adressobjekt	Inhalt
REDE	Subnetz 192.168.2.0/24 (semantischer Alias für den Rede-DMZ-Bereich; aus DMZ_net ausgegliedert für die Rede-spezifischen Policies)
REDE_MAINFRAME	Subnetz 46.224.39.0/24 externer Rede-Node, Gegenstelle für die WireGuard-Sync
lan	Interface-Subnetz 192.168.1.253/24 (automatisch vom LAN-Interface)

3.7 CUSTOM SERVICES

Vom Default abweichend sind folgende eigene Service-Objekte definiert:

Service	Protokoll / Port	Verwendung
rede_traffic	TCP **9443**	Eingehender Rede-Traffic vom WAN, wird via VIP VIP_rede auf 192.168.2.12:9377 weitergeleitet
REDE_WIREGUARD	UDP **51820**	WireGuard-Sync zwischen den Rede-Nodes (DMZ → externer Rede-Node REDE_MAINFRAME)

3.8 SSL-VPN

Benutzer

Benutzer	Passwort	E-Mail	Mitgliedschaft
vpn-keanu	*Passwort im KeePass*	keanu.amann@hotmail.com	SSLVPN-Users
vpn-david	*Passwort im KeePass*	david.kleinfercher@gmail.com	SSLVPN-Users

Einstellungen

Parameter	**Wert**
Port	8443
Zugriff	https://94.199.169.148:8443
Zertifikat	Fortinet_Factory
DNS für Clients	192.168.1.2 (AD), 94.199.168.1 (extern)
Tunnel-IP-Pool	SSLVPN_TUNNEL_ADDR1 (10.212.134.200 10.212.134.210)
Default-Portal	No-Access

```
config vpn ssl settings
    set servercert "Fortinet_Factory"
    set tunnel-ip-pools "SSLVPN_TUNNEL_ADDR1"
    set dns-server1 192.168.1.2
    set dns-server2 94.199.168.1
    set port 8443
    set source-interface "wan1"
    set source-address "all"
    set default-portal "No-Access"
    config authentication-rule
        edit 1
            set groups "SSLVPN-Users"
            set portal "RDP-Portal"
        next
    end
end
```

Portal-Konfiguration

Benutzer der Gruppe SSLVPN-Users werden dem Portal RDP-Portal zugewiesen. Alle anderen erhalten No-Access.

```
edit "RDP-Portal"
    set tunnel-mode enable
    set ip-pools "SSLVPN_TUNNEL_ADDR1"
    set split-tunneling disable
next
```

Fehlerbehebung: VPN ohne LAN-Zugriff

Problem: Ursprünglich war Split-Tunneling aktiviert. VPN-Clients konnten dadurch keine Verbindung zu internen Ressourcen aufbauen, weil der Traffic nicht durch den Tunnel geroutet wurde.

Lösung: Split-Tunneling wurde komplett deaktiviert der gesamte Traffic der VPN-Clients läuft jetzt durch den Tunnel.

Nicht verwendete Alternative: Split-Tunneling mit expliziten Routing-Adressen für 192.168.1.0/24 und 192.168.2.0/24.

3.9 FIREWALL-POLICIES

Policies werden in der Reihenfolge ihrer ID abgearbeitet; die erste passende Policy greift. Aktuell aktive IDs: **1, 5, 7, 8, 9, 10, 11, 12, 13**.

ID	Name	Source	Destination	Schedule	Service	Action	NAT	Security Profiles	Log	Bytes
hierkommtsrein (wan1) → lan2 2										
12	WEB_WAN_TO_DMZ	all	WEB_VIP_HTTP WEB_VIP_HTTPS	always	HTTP HTTPS	ACCEPT	Enabled	no-inspection	UTM	4.95 GB
10	Forwarding_rede_DMZ	all	VIP_rede	always	rede_traffic	ACCEPT	Enabled	no-inspection	UTM	0 B
lan → hierkommtsrein (wan1) 1										
1	lan_to_wan	all	all	always	ALL	ACCEPT	Enabled	no-inspection	UTM	49.32 GB
lan → lan2 1										
8	LAN_to_DMZ	all	DMZ_net	always	SSH HTTP HTTPS PING Veeam Backup Ports	ACCEPT	Disabled	no-inspection	UTM	115.34 MB
lan2 → hierkommtsrein (wan1) 2										
13	REDE_TO_MAINFRAME	REDE	REDE_MAINFRAME	always	REDE_WIREGUARD	ACCEPT	Enabled	no-inspection	UTM	141.70 MB
7	DMZ_to_WAN	DMZ_net	all	always	ALL	ACCEPT	Enabled	no-inspection	UTM	1.46 GB
SSL-VPN tunnel interface (ssl.root) → lan 2										
5	VPN-to-lan	SSLVPN_TUNNEL_ADDR1 SSLVPN-Users	LAN_net	always	RDP HTTPS PING SSH	ACCEPT	Disabled	no-inspection	All	219.89 MB
11	VPN before VIP	all SSLVPN-Users	lan	always	ALL	ACCEPT	Enabled	no-inspection	All	7.61 GB
SSL-VPN tunnel interface (ssl.root) → lan2 1										
9	VPN zu DMZ	all SSLVPN-Users	DMZ_net	always	ALL	ACCEPT	Enabled	no-inspection	UTM	84.43 MB
Implicit 1										
0	Implicit Deny	all	all	always	ALL	DENY			Disab...	268.78 MB

Policy 1 LAN zu WAN (Internet-Zugang)

Ermöglicht allen Geräten im LAN den ausgehenden Zugang ins Internet (mit NAT). Das ist die Standard-Internet-Policy für das interne Netz.

Policy 5 VPN-to-lan

Hauptpolicy für VPN-Benutzer; erlaubt RDP, HTTPS, PING und SSH.

Policy 7 DMZ_to_WAN

Ermöglicht der DMZ den Zugang zum Internet (Updates etc.).

Policy 8 LAN_to_DMZ

Erlaubt dem LAN Zugriff auf die DMZ für Administration (SSH) und Web-Zugriff.

Policy 9 VPN zu DMZ

VPN zu DMZ Erlaubt VPN-Benutzern (Gruppe SSLVPN-Users) den Zugriff auf die Server in der DMZ (ssl.root → lan2, alle Dienste, mit NAT). Damit können Keanu und David nach dem VPN-Login auch die DMZ-Hosts (Web- und Rede-Server) administrieren, nicht nur das LAN.

Nimmt eingehenden Rede-Traffic von außen entgegen und leitet ihn via VIP `VIP_rede` auf den Rede-Relay-Host in der DMZ.

Policy 11 VPN before VIP

Zusätzliche Policy für VPN-Zugriff auf das LAN mit NAT.

Policy 12 WEB_WAN_TO_DMZ

Leitet eingehenden Web-Traffic von außen auf den Webserver in der DMZ (verwendet die VIPs `WEB_VIP_HTTP` und `WEB_VIP_HTTPS`).

Policy 13 REDE_TO_MAINFRAME (WireGuard-Sync nach extern)

Erlaubt der Rede-Node in der DMZ, mit dem externen Rede-Mainframe per WireGuard zu synchronisieren.

Die vollständige, live ausgelesene FortiGate-Konfiguration (inkl. aller Interfaces, VIPs, Services und Zertifikate) liegt als “Volldatei” in der Projektmappe unter *Sonstiges*.

3.10 VIRTUAL IPS (PORT-FORWARDING)

IPv4 Virtual IP 3		
WEB_VIP_HTTP	94.199.169.148 → 192.168.2.11 (TCP: 80 → 80)	hierkommtsrein (wan1)
WEB_VIP_HTTPS	94.199.169.148 → 192.168.2.11 (TCP: 443 → 443)	hierkommtsrein (wan1)
VIP_rede	94.199.169.148 → 192.168.2.12 (TCP: 9443 → 9377)	any

WEB_VIP_HTTP - Web (HTTP)

Eingehender HTTP-Traffic auf 94.199.169.148:80 → Webserver 192.168.2.11:80 in der DMZ.

WEB_VIP_HTTPS - Web (HTTPS)

Eingehender HTTPS-Traffic auf 94.199.169.148:443 → Webserver 192.168.2.11:443 in der DMZ.

VIP_rede | Rede-Relay-Traffic

Eingehender Rede-Traffic auf 94.199.169.148:9443 → Rede-Relay 192.168.2.12:9377 in der DMZ. Der externe Port (9443) wird auf den internen Listening-Port des Rede-Relays (9377) gemappt.

4. Windows-Server (Active Directory)

Das Active Directory bildet das zentrale Rückgrat der Benutzer- und Geräteverwaltung im Projekt. Statt Konten, Rechte und Richtlinien auf jedem System einzeln zu pflegen, werden sie an einer Stelle, dem Domänencontroller winsrvmgmt1, definiert und automatisch auf alle Domänenmitglieder ausgerollt. Das ist die Grundlage dafür, dass sich Benutzer mit denselben Anmeldedaten domänenweit anmelden, dass Berechtigungen über Gruppen statt über Einzelzuweisungen gesteuert werden und dass sich Sicherheitsvorgaben (Passwort-Richtlinien, Account-Lockout, erlaubte Protokolle) zentral per Group Policy durchsetzen lassen.

Der Domänencontroller vereint dabei drei Kernrollen: **Active Directory** (Authentifizierung und Verzeichnis), **DNS** (Namensauflösung für die Domäne t12.lan, Voraussetzung für die AD-Funktion) und **DHCP** (automatische Adressvergabe im LAN, inklusive der PXE-Boot-Optionen für das Baramundi-Deployment). Diese Zentralisierung ist auch die Voraussetzung für die übrigen Dienste im Netz: Baramundi nutzt AD-Service-Konten für seine Client-Verwaltung, Veeam sichert die Domänencontroller als kritisches System, und der Naemon-Host ist per realm/sssd in dieselbe Domäne eingebunden.

4.1 DOMAIN & STANDARD-KONTEN

Feld	**Wert**
Domain	t12.lan
Domain-Administrator	Administrator / *Passwort im KeePass*
Standard-Passwort (neue Benutzer)	*Passwort im KeePass*
Default-Administrator	gelöscht

4.2 BENUTZER

Benutzer	**Beschreibung**	**Passwort**
admindak	Admin-User für Daniel	*Passwort im KeePass*
adminkan	Admin-User für Kanu	*Passwort im KeePass*
barainst	Baramundi Installation (Admins)	*Passwort im KeePass*
baraadmin	Baramundi Admin	*Passwort im KeePass*
baranet	Baramundi Network	*Passwort im KeePass*
Daniel Kleinfercher	Benutzer	*Passwort im KeePass*
Kanu Amann	Benutzer	*Passwort im KeePass*
kairos	Naemon-User, kein Login	*Passwort im KeePass*
backy	Backup-User, kein Login	*Passwort im KeePass*
maintainer	Debian-CT-Login per SSH außerhalb vom LAN	*Passwort im KeePass*
LocalAdmin	Local-Admin auf Clients	*Passwort im KeePass*
donjon	Domain-Join-Rechte, OU t12 Computers	*Passwort im KeePass (Serviceuser)*

4.3 GRUPPEN

Usergruppen

- Admins: admindak, adminkan, backy, baraadmin, Baramundi-Inst, baranet, kairos
- t12Users: dak, kan

Rechtegruppen

- BaraUsers: admindak, dak, kan, adminkan
- RemoteAllowedUsers: admindak, adminkan
- Queryusers: backy, kairos

Computergruppen

- Servers: winsrvbara
- Client Devices: (Clients)

4.4 OU-STRUKTUR

```
t12 Computers      (OU für alle Domänencomputer)
t12 Groups
├ Computer Groups
|   ├ Servers
|   └ Client Devices
├─ User Groups
|   ├ t12Users
|   └ Admins
└-- Permission Groups
    ├ Queryusers
    ├ RemoteAllowedUsers
    └ BaraUsers
t12 Users        (OU für alle Domänenbenutzer)
```

4.5 GROUP POLICY OBJECTS

Bei `t12 Computers`:

Allowed Remote Access

Security Filtering

The settings in this GPO can only apply to the following groups, users, and computers:

Name
NT AUTHORITY\Authenticated Users
T12\RemoteAllowedUsers

Delegation

These groups and users have the specified permission for this GPO

Name	Allowed Permissions	Inherited
NT AUTHORITY\Authenticated Users	Read (from Security Filtering)	No
NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS	Read	No
NT AUTHORITY\SYSTEM	Edit settings, delete, modify security	No
T12\admindak	Edit settings, delete, modify security	No
T12\Domain Admins	Edit settings, delete, modify security	No
T12\Enterprise Admins	Edit settings, delete, modify security	No
T12\RemoteAllowedUsers	Read (from Security Filtering)	No

Computer Configuration (Enabled)

Policies

Windows Settings

Security Settings

Local Policies/User Rights Assignment

Policy	Setting
Allow log on through Terminal Services	T12\RemoteAllowedUsers

Administrative Templates

Policy definitions (ADMX files) retrieved from the local computer.

Windows Components/Remote Desktop Services/Remote Desktop Session Host/Connections

Policy	Setting	Comment
Allow users to connect remotely by using Remote Desktop Services	Enabled	
Limit number of connections	Enabled	
RD Maximum Connections allowed Type 999999 for unlimited connections.	5	

User Configuration (Enabled)

No settings defined.

Block insecure Protocols

Security Filtering

The settings in this GPO can only apply to the following groups, users, and computers:

Name
NT AUTHORITY\Authenticated Users

Delegation

These groups and users have the specified permission for this GPO

Name	Allowed Permissions	Inherited
NT AUTHORITY\Authenticated Users	Read (from Security Filtering)	No
NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS	Read	No
NT AUTHORITY\SYSTEM	Edit settings, delete, modify security	No
T12\Domain Admins	Edit settings, delete, modify security	No
T12\Enterprise Admins	Edit settings, delete, modify security	No

Computer Configuration (Enabled)

Policies

Windows Settings

Security Settings

Local Policies/Security Options

Domain Controller

Policy	Setting
Domain controller: LDAP server signing requirements	Require signing

Network Access

Policy	Setting
Network access: Do not allow anonymous enumeration of SAM accounts and shares	Enabled
Network access: Let Everyone permissions apply to anonymous users	Disabled

Network Security

Policy	Setting
Network security: Do not store LAN Manager hash value on next password change	Enabled
Network security: LAN Manager authentication level	Send NTLMv2 response only. Refuse LM & NTLM

Other

Policy	Setting
Network security: Restrict NTLM: Audit Incoming NTLM Traffic	Enable auditing for domain accounts
Network security: Restrict NTLM: Audit NTLM authentication in this domain	Enable for domain accounts

Administrative Templates

Policy definitions (ADMX files) retrieved from the local computer.

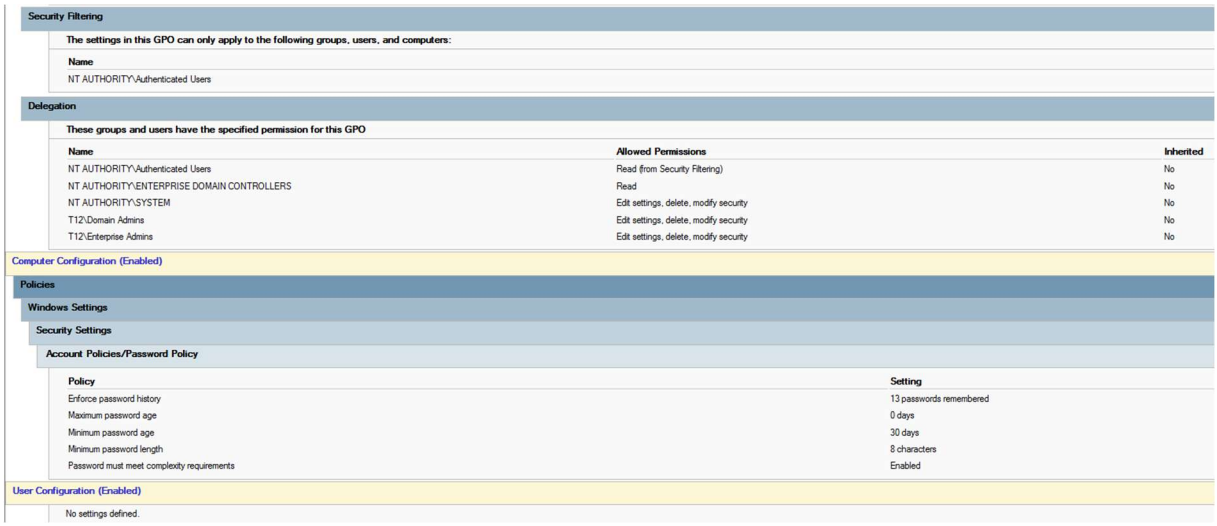
Network/Lanman Workstation

Policy	Setting	Comment
Enable insecure guest logons	Disabled	

User Configuration (Enabled)

No settings defined.

Password Policies



Security Filtering

The settings in this GPO can only apply to the following groups, users, and computers:

Name
NT AUTHORITY\Authenticated Users

Delegation

These groups and users have the specified permission for this GPO

Name	Allowed Permissions	Inherited
NT AUTHORITY\Authenticated Users	Read (from Security Filtering)	No
NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS	Read	No
NT AUTHORITY\SYSTEM	Edit settings, delete, modify security	No
T12\Domain Admins	Edit settings, delete, modify security	No
T12\Enterprise Admins	Edit settings, delete, modify security	No

Computer Configuration (Enabled)

Policies

Windows Settings

Security Settings

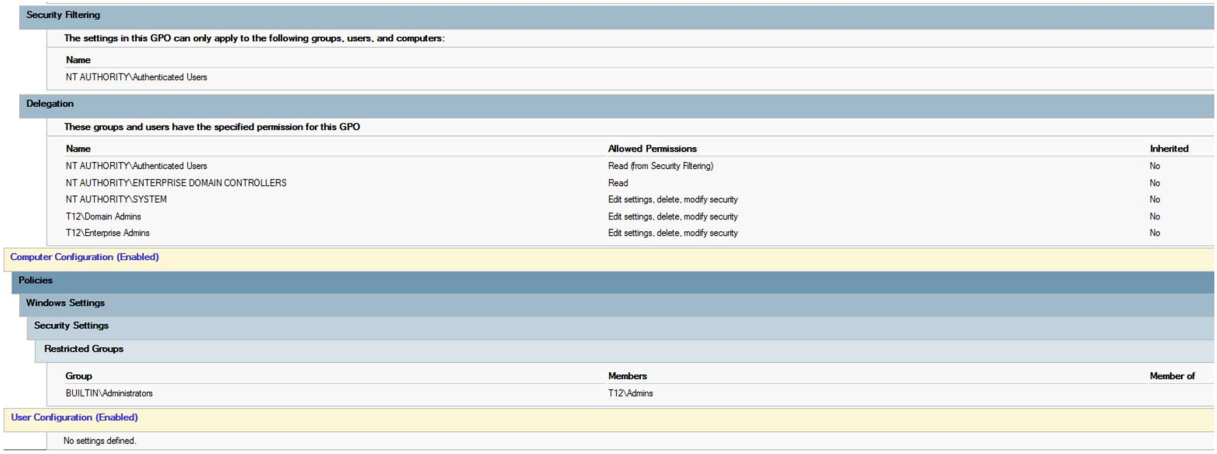
Account Policies/Password Policy

Policy	Setting
Enforce password history	13 passwords remembered
Maximum password age	0 days
Minimum password age	30 days
Minimum password length	8 characters
Password must meet complexity requirements	Enabled

User Configuration (Enabled)

No settings defined.

Restricted Groups



Security Filtering

The settings in this GPO can only apply to the following groups, users, and computers:

Name
NT AUTHORITY\Authenticated Users

Delegation

These groups and users have the specified permission for this GPO

Name	Allowed Permissions	Inherited
NT AUTHORITY\Authenticated Users	Read (from Security Filtering)	No
NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS	Read	No
NT AUTHORITY\SYSTEM	Edit settings, delete, modify security	No
T12\Domain Admins	Edit settings, delete, modify security	No
T12\Enterprise Admins	Edit settings, delete, modify security	No

Computer Configuration (Enabled)

Policies

Windows Settings

Security Settings

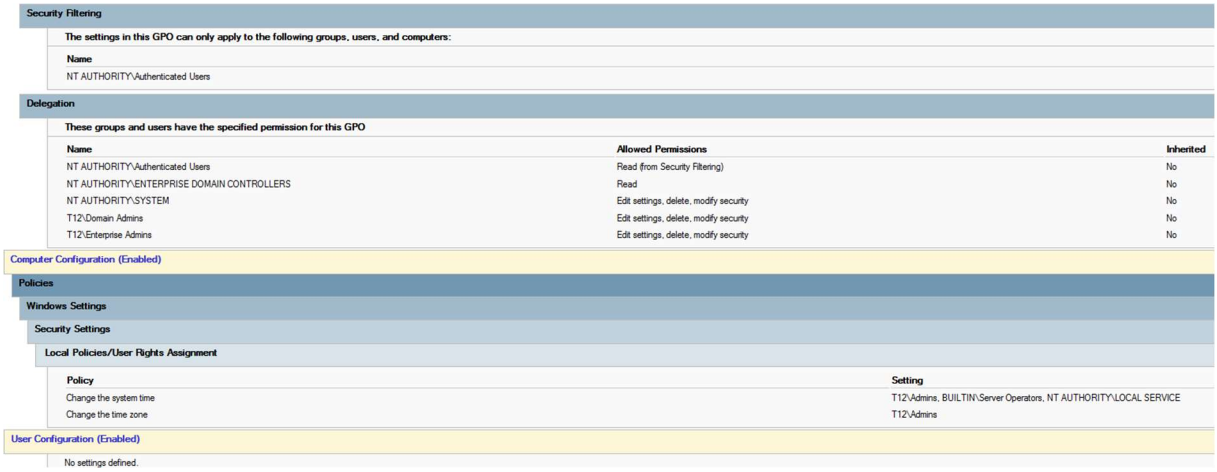
Restricted Groups

Group	Members	Member of
BUILTIN\Administrators	T12\Admins	

User Configuration (Enabled)

No settings defined.

Time Sync



Security Filtering

The settings in this GPO can only apply to the following groups, users, and computers:

Name
NT AUTHORITY\Authenticated Users

Delegation

These groups and users have the specified permission for this GPO

Name	Allowed Permissions	Inherited
NT AUTHORITY\Authenticated Users	Read (from Security Filtering)	No
NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS	Read	No
NT AUTHORITY\SYSTEM	Edit settings, delete, modify security	No
T12\Domain Admins	Edit settings, delete, modify security	No
T12\Enterprise Admins	Edit settings, delete, modify security	No

Computer Configuration (Enabled)

Policies

Windows Settings

Security Settings

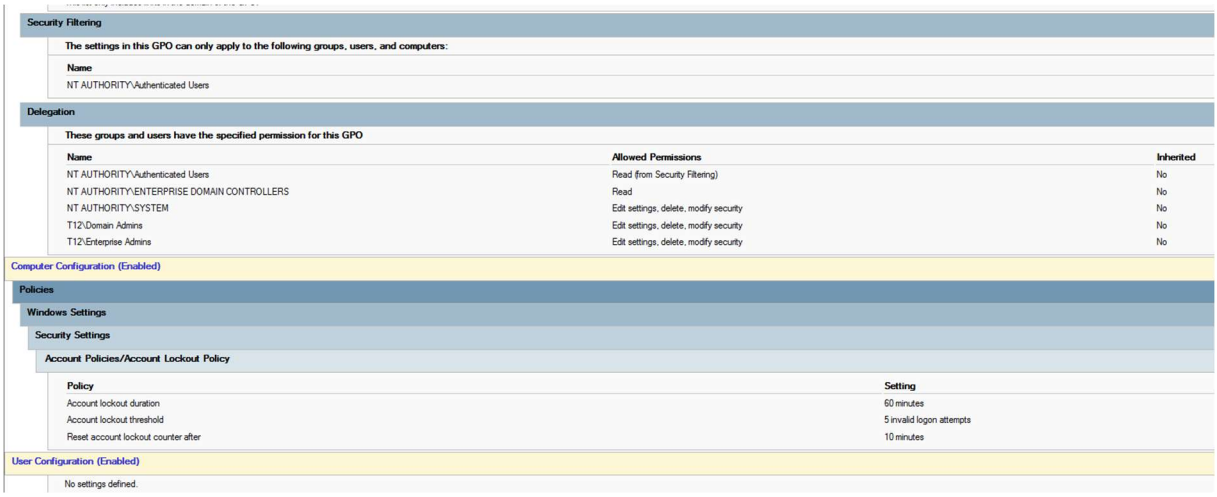
Local Policies/User Rights Assignment

Policy	Setting
Change the system time	T12\Admins, BUILTIN\Server Operators, NT AUTHORITY\LOCAL SERVICE
Change the time zone	T12\Admins

User Configuration (Enabled)

No settings defined.

Account Lockout Policies



Security Filtering

The settings in this GPO can only apply to the following groups, users, and computers:

Name
NT AUTHORITY\Authenticated Users

Delegation

These groups and users have the specified permission for this GPO

Name	Allowed Permissions	Inherited
NT AUTHORITY\Authenticated Users	Read (from Security Filtering)	No
NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS	Read	No
NT AUTHORITY\SYSTEM	Edit settings, delete, modify security	No
T12\Domain Admins	Edit settings, delete, modify security	No
T12\Enterprise Admins	Edit settings, delete, modify security	No

Computer Configuration (Enabled)

Policies

Windows Settings

Security Settings

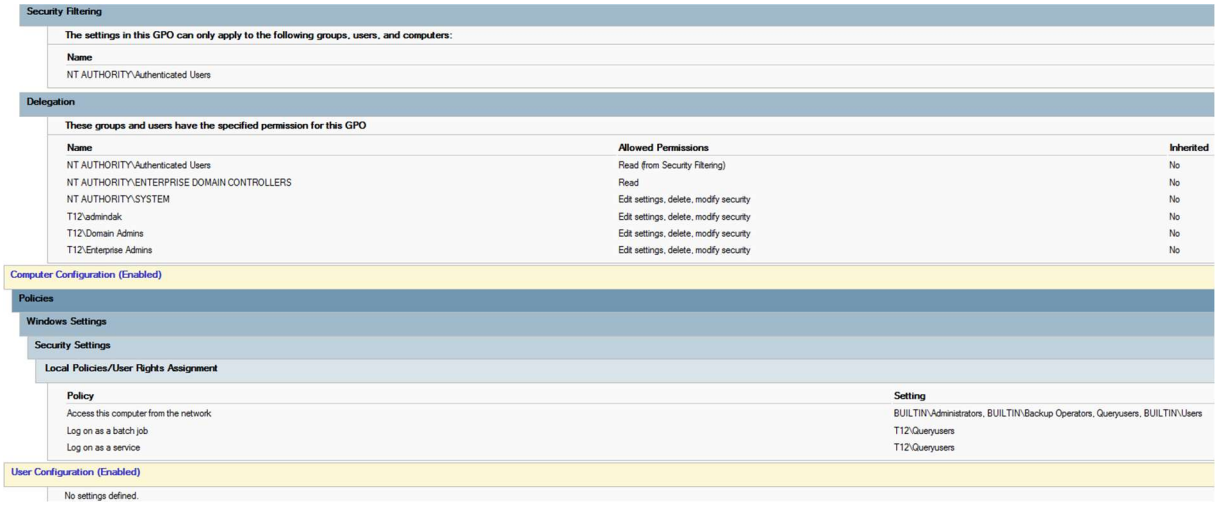
Account Policies/Account Lockout Policy

Policy	Setting
Account lockout duration	60 minutes
Account lockout threshold	5 invalid logon attempts
Reset account lockout counter after	10 minutes

User Configuration (Enabled)

No settings defined.

Sign on as a Service



Security Filtering

The settings in this GPO can only apply to the following groups, users, and computers:

Name
NT AUTHORITY\Authenticated Users

Delegation

These groups and users have the specified permission for this GPO

Name	Allowed Permissions	Inherited
NT AUTHORITY\Authenticated Users	Read (from Security Filtering)	No
NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS	Read	No
NT AUTHORITY\SYSTEM	Edit settings, delete, modify security	No
T12\admindak	Edit settings, delete, modify security	No
T12\Domain Admins	Edit settings, delete, modify security	No
T12\Enterprise Admins	Edit settings, delete, modify security	No

Computer Configuration (Enabled)

Policies

Windows Settings

Security Settings

Local Policies/User Rights Assignment

Policy	Setting
Access this computer from the network	BUILTIN\Administrators, BUILTIN\Backup Operators, Queryusers, BUILTIN\Users
Log on as a batch job	T12\Queryusers
Log on as a service	T12\Queryusers

User Configuration (Enabled)

No settings defined.

4.6 DHCP

Parameter	**Wert**
Scope	192.168.1.20 - 192.168.1.240
Gateway	192.168.1.253 (Firewall)
DNS	192.168.1.2 (winsrvmgmt1)

Zusätzlich für Baramundi-PXE-Boot:

```
Set-DhcpServerv4OptionValue -ScopeId 192.168.1.0 -OptionId 66 -Value "192.168.1.4"
Set-DhcpServerv4OptionValue -ScopeId 192.168.1.0 -OptionId 67 -Value "BMSNBP.PXE"
```

4.7 DNS

Parameter	**Wert**
Forward-Lookup-Zone	T12DNSZONE mit Secure Dynamic Updates
Forwarder	94.199.168.1, 94.199.169.1

5. Server im LAN

5.1 WINSRVMGMT1 MANAGEMENT

Der Management-Server ist der zentrale Domänencontroller und das Fundament des gesamten LAN. Er vereint die drei Infrastruktur-Rollen, von denen praktisch alle anderen Systeme abhängen: Active Directory (zentrale Authentifizierung und Verzeichnis), DNS (Namensauflösung für t12.lan, ohne die AD nicht funktioniert) und DHCP (automatische IP-Vergabe im LAN samt der PXE-Boot-Optionen für das Baramundi-Deployment). Fällt dieser Host aus, können sich Benutzer nicht mehr anmelden und die Namensauflösung im Netz bricht weg, entsprechend wird er von Veeam als kritisches System gesichert (eigener Backup-Job, siehe §5.2).

Feld	**Wert**
IP	192.168.1.2
Funktionen	Active Directory, DNS, DHCP

5.2 WINSRVBACKUP1 VEEAM BACKUP

Dieser Server ist die zentrale Backup-Instanz des Projekts. Auf ihm läuft Veeam Backup & Replication, das die wichtigsten Systeme (Domänencontroller, Baramundi-, Naemon- und Rede-Server) nach einem festen Zeitplan als Voll-Backups auf das Laufwerk B: sichert. Hintergrund ist die eingeschränkte, zweckentfremdete Hardware der Proxmox-Hosts (siehe §2): Da ein Hardware-Ausfall realistisch eingeplant werden muss, sorgen die Backups dafür, dass einzelne Systeme im Fehlerfall gezielt wiederhergestellt werden können. Der Server ist als einziger auf dem zweiten Proxmox-Knoten untergebracht, um Backups physisch von der produktiven Infrastruktur zu trennen. Die vollständige Backup-Strategie ist in der Backup Dokumentation beschrieben.

Feld	**Wert**
IP	192.168.1.9
Domäne	t12.lan
Software	Veeam
Backup-Volume	Laufwerk B:
Service-User	backy@t12.lan (*Passwort im KeePass*)

Veeam-Jobs:

Job	**Ziel(e)**	**Retention**	**Typ**	**Plan**
Backup Baramundi Server	winsrvbara1	14 Tage	Full	Samstag 20:00
Backup Domain Controllers	winsrvmgmt1	14 Tage	Full	Freitag 20:00
Backup Linux	ubusrvnaemon1, debsrvrede1	14 Tage	Full	Sonntag 20:00

5.3 WINSRVBARA1 BARAMUNDI

Der Baramundi-Server übernimmt das zentrale Client-Management und Software-Deployment. Über die Baramundi Management Suite lassen sich Endgeräte automatisiert betanken (OS-Installation per PXE-Boot), mit Software versorgen und verwalten, ohne jeden Client manuell anfassen zu müssen. Der Server nutzt dafür dedizierte AD-Service-Konten (baraadmin, barainst, baranet) und eine lokale SQL-Server-Express-Instanz als Datenbank-Backend. Wegen der engen Verzahnung von Diensten (bServer, SQL, Lizenz-Management) ist dieser Host vergleichsweise wartungsintensiv, die aufgetretenen Probleme und ihre Lösungen sind in §9 dokumentiert.

Feld	**Wert**
IP	192.168.1.4/24
DNS	192.168.1.2
OS	Windows Server 2022 (4 Kerne) v10.0.20348
Domäne	T12.lan
Administrator	*Passwort im KeePass*

Feld	**Wert**
Produkt	Baramundi Management Suite (bMS)
Version	2025 R2 (25.2.130.0)
Installationsverzeichnis	C:\Program Files(x86)\baramundi\Management Server\
Konfigurationsdatei	C:\Program Files (x86)\baramundi\Management Server\baramundi.config
Log-Verzeichnis	C:\ProgramData\baramundi\Logs\
bServer-Logdatei (täglich)	C:\ProgramData\baramundi\Logs\bserver_YYYY-MM-DD.0.log

Feld	Wert
DB-Verwaltungstool	C:\Program Files (x86)\baramundi\Management Server\DBMgrCmd.exe
Management Center (bMC)	C:\Program Files (x86)\baramundi\Management Center\bMC.exe

Standardinstallation mit Setup, Baramundi Automation Studio und Server. Zusätzlich PowerShell 7 installiert.

Baramundi-Dienste

Dienst	Starttyp	Funktion
`bLicenseManagement`	Automatic	Lizenz-Verwaltung
`bServer`	Automatic (Delayed)	Kern-Dienst, läuft als `NT AUTHORITY\SYSTEM`
`bWebserver`	Automatic	Web-Frontend

Endpunkte / Ports

Endpoint	URL / Port
HTTP MOC	`http://localhost:10081`
HTML View	`http://localhost:50313`
bMS Connector	TCP `10091`

SQL-Server

Parameter	Wert
Edition	SQL Server 2022 Express
Instanz	`.\SQLEXPRESS`
Datenbank	`baraSQL`
Connection String	`Server=localhost\SQLEXPRESS;Database=master;Trusted_Connection=True;`
Authentifizierung	Windows-Auth (Trusted Connection) – kein separates SQL-Passwort
Install-Log-Folder	`C:\Program Files\Microsoft SQL Server\160\Setup Bootstrap\Log\20260303_145358`
Installation Media	`C:\SQL2022\Express_ENU`

Datenbank-Berechtigung für `bServer` (Pflicht für Erstinstallation): Der Dienst `bServer` läuft als `NT AUTHORITY\SYSTEM`. Dieses Konto muss explizit Zugriff auf `baraSQL` bekommen, sonst startet der Dienst nicht (siehe §9, Problem 2).

```
USE baraSQL;
CREATE USER [NT AUTHORITY\SYSTEM] FOR LOGIN [NT AUTHORITY\SYSTEM];
ALTER ROLE db_owner ADD MEMBER [NT AUTHORITY\SYSTEM];
```

SQL Server Agent aktivieren (optional, für geplante Wartungsaufgaben):

```
Set-Service -Name 'SQLAgent$SQLEXPRESS' -StartupType Automatic
Start-Service 'SQLAgent$SQLEXPRESS'
```

Service-Start-Hardening

Zwei Anpassungen, damit `bServer` auch nach einem Server-Neustart zuverlässig hochkommt:

Problem	**Lösung**
SQL Browser deaktiviert → `bServer` konnte `.\SQLEXPRESS` nicht finden	SQL Browser-Dienst auf Automatic gesetzt und gestartet
Windows Service Timeout zu kurz (30 s) → `bServer` lädt länger	`ServicesPipeTimeout` auf 120 000 ms (120 s) in der Registry erhöht

```
# SQL Browser dauerhaft aktivieren
Set-Service -Name 'SQLBrowser' -StartupType Automatic
Start-Service 'SQLBrowser'

# Service-Start-Timeout auf 120 s erhöhen
New-ItemProperty `
    -Path 'HKLM:\SYSTEM\CurrentControlSet\Control' `
    -Name 'ServicesPipeTimeout' `
    -PropertyType DWord `
    -Value 120000 `
    -Force
# Änderung wird nach Neustart wirksam
```

PXE-Firewall-Regeln

Drei eingehende Regeln für die Baramundi-OS-Install-Funktion:

Regel-Name	Richtung	Protokoll	Port	Zweck
`baramundi PXE - TFTP`	Inbound	UDP	69	TFTP-Dateiübertragung (Boot-Image)
`baramundi PXE - Proxy DHCP`	Inbound	UDP	4011	PXE Proxy DHCP (Boot-Optionen)
`baramundi PXE - DHCP Relay`	Inbound	UDP	67–68	DHCP-Relay für PXE-Clients

```
New-NetFirewallRule -DisplayName "baramundi PXE - TFTP" `
    -Direction Inbound -Protocol UDP -LocalPort 69 `
    -Action Allow -Profile Domain,Private

New-NetFirewallRule -DisplayName "baramundi PXE - Proxy DHCP" `
    -Direction Inbound -Protocol UDP -LocalPort 4011 `
    -Action Allow -Profile Domain,Private

New-NetFirewallRule -DisplayName "baramundi PXE - DHCP Relay" `
    -Direction Inbound -Protocol UDP -LocalPort 67-68 `
    -Action Allow -Profile Domain,Private
```

> Ergänzt die DHCP-Server-Optionen 66/67, die in §4.6 (DHCP) für den PXE-Boot gesetzt werden.

5.4 UBUSRVNAEMON1 NAEMON MONITORING

Der Naemon-Host ist der Monitoring-Server des Projekts. Er überwacht kontinuierlich die Erreichbarkeit und den Zustand aller wichtigen Systeme (CPU, RAM, Storage, Uptime, Event-Logs) und schlägt Alarm, bevor aus einem Engpass ein Ausfall wird. Gerade wegen der leistungsschwachen, zweckentfremdeten Hardware (siehe §2) ist dieses frühzeitige Erkennen von Überlastungen der Hauptgrund, überhaupt Monitoring zu betreiben. Der Host ist per realm/sssd in die Domäne t12.lan eingebunden und nutzt den Query-User kairos für seine Abfragen. Die vollständige Monitoring-Konfiguration (Plugins, Checks, Thruk-Dashboard) ist in der Monitoring Dokumentation beschrieben.

Feld	**Wert**
IP	192.168.1.11
root / user	*Passwort im KeePass* (gleich für beide Konten)
Domain-Join	über realm + sssd an t12.lan
Erlaubte Gruppen	Queryuser, Admins

Installierte Software

- vim
- python3 mit pyYAML, pip und flask
- naemon
- thruk
- aiowmi WMIC-Server
- check_wmi_plus (Naemon-Plugin)

> **Detaillierte Dokumentation:** Naemon-Konfiguration, Plugins, Thruk-Setup und WMIC-Server sind im Dokument “Monitoring Dokumentation” ausführlich beschrieben. Die folgenden Abschnitte sind die Kurzform.

/etc/naemon/resource.cfg

```
# Sets variables to be the path to the plugins
$USER1$=/usr/lib/naemon/plugins
$USER2$=/usr/lib/nagios/plugins
# Sets $USER2$ to be the path to event handlers
#$USER2$=/usr/lib/naemon/plugins/eventhandlers

# Store some usernames and passwords (hidden from the CGIs)
$USER4$=user1
$USER5$=<Token im KeePass>            # WMIC-Server-Access-Token
$USER6$=kairos
$USER7$=/etc/naemon/sshpass.secret    # SSH-Passwort von kairos (im KeePass)
```

/etc/naemon/conf.d/ Übersicht

- printer.cfg wurde **gelöscht**.
- switch.cfg wurde in **`firewalls.cfg`** umbenannt.

windows.cfg

```
    define host {
  host_name       winsrvmgmt1
  alias         AD, DHCP and DNS Server
  address       192.168.1.2
  use           windows-server
  hostgroups      windows-servers
}

define host {
  host_name       winbarasrv1
  alias         Baramundiserver
  address       192.168.1.4
  use           windows-server
  hostgroups      windows-servers
}

define host {
  host_name       winsrvbackup1
  alias         Backupserver
  address       192.168.1.9
  use           windows-server
  hostgroups      windows-servers
}
###############################################################################
# HOST GROUP DEFINITIONS
###############################################################################

define hostgroup {
  hostgroup_name    windows-servers
  alias         Windows Servers
}

###############################################################################
# SERVICE DEFINITIONS
###############################################################################
define service {
  service_description Uptime
  hostgroup_name    windows-servers
  use           generic-service
  check_command     check-uptime
}
```

```
define service {
  service_description Ping
  hostgroup_name    windows-servers
  use           generic-service
  check_command     check-host-alive
}

define service {
  service_description Event_View
  hostgroup_name    windows-servers
  use           generic-service
  check_command     check-events
}

define service {
  service_description CPU_Load
  hostgroup_name    windows-servers
  use           generic-service
  check_command     check-cpu
}

define service {
  service_description RAM_Load
  hostgroup_name    windows-servers
  use           generic-service
  check_command     check-ram
}

define service {
  service_description Storage_Use
  hostgroup_name    windows-servers
  use           generic-service
  check_command     check-storage
}
```

commands.cfg Custom Commands

```
###############################################################################
# CUSTOM COMMANDS  t12.lan
###############################################################################

#-------------------------------------------------------------------
# Windows Check Commands
#-------------------------------------------------------------------

define command {
  command_name     check-storage
  command_line     $USER1$/check_wmi_plus.pl -H $HOSTADDRESS$ -m checkdrivesize -a "C:" -w 75 -c 90 -u $USER4$ -p
$USER5$
}

define command {
  command_name     check-cpu
  command_line     $USER1$/check_wmi_plus.pl -H $HOSTADDRESS$ -m checkcpu -w 80 -c 95 -u $USER4$ -p $USER5$
}

define command {
  command_name     check-ram
  command_line     $USER1$/check_wmi_plus.pl -H $HOSTADDRESS$ -m checkmem -w 80 -c 90 -u $USER4$ -p $USER5$
}

define command {
  command_name     check-events
  command_line     $USER1$/check_wmi_plus.pl -H $HOSTADDRESS$ -m checkeventlog -a "System" --includedata
"_EventType=1" --timebefore 86400 -w 0 -c 0 -u $USER4$ -p $USER5$
}

define command {
  command_name     check-uptime
  command_line     $USER1$/check_wmi_plus.pl -H $HOSTADDRESS$ -m checkuptime -u $USER4$ -p $USER5$
}

#-------------------------------------------------------------------
# Linux Check Commands
#-------------------------------------------------------------------

define command {
  command_name     check-linux-storage
  command_line     sshpass -f $USER7$ $USER2$/check_by_ssh -H $HOSTADDRESS$ -l $USER6$ -o StrictHostKeyChecking=no -C
"/usr/lib/nagios/plugins/check_disk -w 25% -c 10%"
}

define command {
  command_name     check-linux-ram
  command_line     sshpass -f $USER7$ $USER2$/check_by_ssh -H $HOSTADDRESS$ -l $USER6$ -o StrictHostKeyChecking=no -C
"/usr/lib/nagios/plugins/check_memory --available -w 20%: -c 10%:"
}
```

```
define command {
  command_name     check-linux-cpu
  command_line     sshpass -f $USER7$ $USER2$/check_by_ssh -H $HOSTADDRESS$ -l $USER6$ -o StrictHostKeyChecking=no -C
"/usr/lib/nagios/plugins/check_load -w 1.5,1.0,0.8 -c 3.0,2.5,2.0"
 }

define command {
  command_name     check-linux-uptime
  command_line     sshpass -f $USER7$ $USER2$/check_by_ssh -H $HOSTADDRESS$ -l $USER6$ -o StrictHostKeyChecking=no -C
"/usr/lib/nagios/plugins/check_uptime"
 }
```

localhost.cfg (Linux)

```
define host {
  host_name       ubunaemonsrv1
  alias         Naemon Server
  address       127.0.0.1
  use           linux-server
  hostgroups      naemon-servers
}

###############################################################################
# HOST GROUP DEFINITION
###############################################################################

define hostgroup {
  hostgroup_name    naemon-servers
  alias         Naemon Servers
}
```

firewall.cfg

```
define host {
  host_name       t12graetefw1       ; Name dieses Geräts (Switch/Firewall)
  alias         Main Firewall       ; Langer Name
  address       192.168.1.253       ; IP-Adresse
  use           generic-switch
  hostgroups      firewalls
}
```

Thruk

Feld	**Wert**
Admin-User	thrukadmin
Passwort	*Passwort im KeePass*

aiowmi / WMIC-Server

Änderungen an wmic_server.yaml:

- Alle Default-User und -Tokens **bis auf** user1 entfernen (Token *im KeePass*).
- Credentials von kairos werden bei user1 ergänzt.

Der WMIC-Server läuft als Service über **gunicorn**.

6. Server in der DMZ

6.1 DEBSRVWEBCT1 WEBSERVER

Der Webserver hostet den öffentlich erreichbaren Webauftritt des Projekts unter https://team12.website. Er ist der einzige Dienst, der bewusst aus dem Internet erreichbar gemacht wird, und steht deshalb in der DMZ, also dem vom internen LAN abgeschirmten Netzsegment. Diese Trennung ist Absicht: Würde der Server kompromittiert, hätte ein Angreifer nur Zugriff auf die DMZ und nicht direkt auf die produktiven Systeme im LAN.

Anders als der Rede-Server (§6.2), der als vollwertige Debian-VM läuft, ist der Webserver bewusst als **leichtgewichtiger LXC-Container** umgesetzt (erkennbar am CT im Namen). Ein Container teilt sich den Kernel mit dem Proxmox-Host und braucht dadurch deutlich weniger CPU, RAM und Speicher als eine vollständige VM. Angesichts der stark begrenzten, zweckentfremdeten Hardware (siehe §2) ist das eine bewusste Entscheidung zur Ressourcenschonung, für einen Webdienst genügt ein Container vollkommen, und die eingesparten Ressourcen stehen den schwereren VMs zur Verfügung.

Der Zugang von außen läuft kontrolliert über die Firewall: Eingehender HTTP-/HTTPS-Traffic auf die öffentliche IP wird per Virtual IP (WEB_VIP_HTTP / WEB_VIP_HTTPS) und Policy 12 gezielt auf diesen Host weitergeleitet (siehe §3.9 und §3.10). Die TLS-Verschlüsselung übernimmt ein Let's-Encrypt-Zertifikat, das per Certbot automatisch ausgestellt und erneuert wird. Als einziger Host nutzt der Webserver bewusst externe DNS-Server (1.1.1.1, 8.8.8.8) statt des internen AD-DNS, um nicht von der internen Namensauflösung abhängig zu sein.

Feld	**Wert**
IP	192.168.2.11
Gateway	192.168.2.253 (Firewall)
root	*Passwort im KeePass*
Installierte Software	npm, git, curl, certbot
DNS	1.1.1.1, 8.8.8.8 (externe DNS statt interne)
TLS	Let's-Encrypt-Zertifikat via Certbot für https://team12.website

Eingehender HTTP/HTTPS-Traffic kommt über die VIPs WEB_VIP_HTTP und WEB_VIP_HTTPS (§3.10) und die Policy WEB_WAN_TO_DMZ (§3.9, Policy 12) hier an.

6.2 DEBSRVREDE1 REDE RELAY

Dieser Host betreibt die Server-Seite von **Rede**, dem selbst entwickelten Ende-zu-Ende-verschlüsselten Messenger des Projekts. Das Relay nimmt die verschlüsselten Nachrichten der Clients entgegen, speichert sie zwischen und stellt sie an die Empfänger zu, es sieht dabei selbst nur Chiffretext und hat zu keinem Zeitpunkt Zugriff auf Klartext-Nachrichten, Schlüssel oder Audiodaten (Details zum Sicherheitsmodell in der Rede-Server- und Rede-Client-Dokumentation). Wie der Webserver steht auch das Relay bewusst in der DMZ, da es aus dem Internet erreichbar sein muss.

Der Zugriff von außen läuft kontrolliert über zwei Wege: Eingehender Client-Traffic erreicht das Relay über die Virtual IP VIP_rede (externer Port 9443 → interner Listening-Port 9377, Policy 10, siehe §3.9/§3.10). Zusätzlich synchronisiert sich das Relay über einen WireGuard-Tunnel (UDP 51820, Policy 13) mit einem externen Rede-Knoten (REDE_MAINFRAME), sodass Nachrichten über Server-Grenzen hinweg föderiert ausgetauscht werden können. Für die Wartung von außerhalb des LAN existiert der SSH-Zugang über den Benutzer maintainer; die Backup-Sicherung erfolgt über backy.

Feld	**Wert**
IP	192.168.2.12
Listening-Port	9377/tcp (Rede-Relay)
Externer Port	9443/tcp (via VIP VIP_rede, §3.10)
root	*Passwort im KeePass*
maintainer	*Passwort im KeePass*
kairos	*Passwort im KeePass*
backy	*Passwort im KeePass*
Installierte Software	vim, nagios-plugins-basic, nagios-plugins-contrib, nginx

Traffic-Pfade:

- **Eingehend (WAN → Relay):** Custom Service rede_traffic (TCP 9443) → Policy 10 Forwarding_rede_DMZ → VIP VIP_rede → 192.168.2.12:9377.
- **Ausgehend (Relay → externer Rede-Mainframe):** Custom Service REDE_WIREGUARD (UDP 51820) → Policy 13 REDE_TO_MAINFRAME → Adressobjekt REDE_MAINFRAME (46.224.39.0/24).

7. Autounattend (Windows-Client-Rollout)

Beim Ausrollen mehrerer Windows-Clients wäre es mühsam und fehleranfällig, jede Maschine von Hand durch das Setup zu klicken, Sprache wählen, Konto anlegen, in die Domäne joinen, Einstellungen setzen. Die **autounattend.xml** automatisiert genau diesen Prozess: Sie wird vom Windows-Setup beim Start eingelesen und beantwortet alle Installationsabfragen automatisch, sodass ein Client ohne manuelle Eingaben vollständig einsatzbereit hochkommt.

Im Projekt greift das mit dem Baramundi-PXE-Boot ineinander: Ein Gerät bootet über das Netzwerk (PXE), zieht das Windows-Image, und die autounattend.xml erledigt den Rest, einheitliche Region/Sprache, ein vordefiniertes lokales Administrator-Konto, die gewünschten Personalisierungs-Einstellungen und vor allem der **automatische Domänenbeitritt** zu t12.lan. Damit landet jeder neue Client reproduzierbar im selben, korrekt konfigurierten Zustand und ist sofort Teil der zentralen AD-Verwaltung (Sektion 4), ohne dass ein Admin daneben sitzen muss.

Besonders relevant ist der Domain-Join-Block: Über den dedizierten Service-User donjon tritt die Maschine automatisiert der Domäne bei und wird direkt in der richtigen Organisationseinheit (OU=t12 Computers) abgelegt, so greifen sofort die passenden Gruppenrichtlinien.

REGION & SPRACHE

- **Home location / Setup-Region:** Austria (Default war US)
- **Tastaturlayout:** German (statt Sprach-Default)

BENUTZERKONTEN

- Lokales Konto angelegt: `LocalAdmin` (Display-Name `LocalAdmin`), Passwort *im KeePass*, Mitglied der Gruppe `Administrators` (statt der Default-Vorgabe `Users`).

PERSONALIZATION

- **Color theme Taskbar/Startmenü:** Dark (Default war Light)
- **Color theme Apps:** Dark (Default war Light)

XML-MARKUP ZUSÄTZLICHE KOMPONENTEN

In der Komponente `Microsoft-Windows-UnattendedJoin` (Pass `specialize`) wurde der Domain-Join-Block ergänzt:

Feld	**Wert**
Domain	`t12.lan`
Service-User	`donjon`
Service-User-Passwort	*Passwort im KeePass*
JoinDomain	`t12.lan`
MachineObjectOU	`OU=t12 Computers,DC=t12,DC=lan`

8. Baramundi-Lizenz

Feld	Wert
Lizenznehmer (Kontakt)	Keanu Amann, Lehrling Benutzerservice, illwerke vkw AG
Hersteller-Kontakt	Agnes Miller, Inside Sales Specialist, baramundi software GmbH (`Agnes.Miller@baramundi.com`)
Lizenztyp	Teststellung (Schul- bzw. Abschlussprojekt)
Aktivierungs-Ticket	`8a3a7265-3dda-4233-bd0d-462cbb498dc5`
Aktivierung	06.05.2026, 09:10 Uhr MESZ
Lizenz gültig bis	30.06.2026 (verlängert am 20.05.2026)
Geplanter Funktionsumfang	Vollzugriff auf alle Module der bMS (Softwareverteilung, automatisierte Installationen, Defense Control, Configuration/PXE)
Beobachteter Funktionsumfang	aktuell nur **Defense Control** und **Configuration/PXE** sichtbar, Jobs lassen sich nicht anlegen (Status 27.05.2026, offen, siehe §9, Problem 3)
Hersteller-Support	Telefon `+49 821 567 08-500` (Mo–Fr 08:30–17:00) · E-Mail `support@baramundi.com`
Aktivierungs-Anleitung	https://www.baramundi.com (Dokumentation)

KORRESPONDENZ-HISTORIE

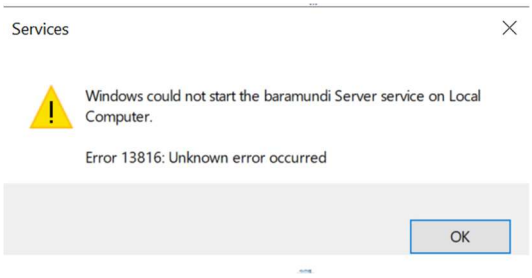
Datum	Richtung	Inhalt (Zusammenfassung)
31.07.2025	Amann → baramundi (`vertrieb@baramundi.com`)	Erstanfrage für eine Teststellung im Rahmen des Abschlussprojekts; Bitte um Testinstallation, Schulungsmaterialien und technische Unterlagen
07.08.2025	Miller (baramundi) → Amann	Zusage zur Teststellung; Hinweis auf kostenpflichtige Schulungen auf `baramundi.com`
08.08.2025	Amann → Miller	Bestätigung des Bedarfs ab 09.09.2025; Bitte um Zugangsdaten / nächste Schritte
11.08.2025	Miller → Amann	Übermittlung der Aktivierungs-Ticketnummer `8a3a7265-…`; Hinweis auf Online-Self-Service für künftige Lizenz- und Funktionserweiterungen; Support-Kontakte
06.05.2026	-	Aktivierung der Lizenz auf `winsrvbara1` erfolgreich (`09:10 Uhr MESZ`)
19.05.2026	Amann → Miller	Rückmeldung: Aktivierung und bMS-Einrichtung problemlos; nur Defense Control + Configuration/PXE sichtbar, keine Jobs anlegbar; Bitte um Prüfung des Lizenzumfangs
20.05.2026	Miller → Amann	Verdacht auf abgelaufene Lizenz; Verlängerung bis 30.06.2026; Aussage "Sie sollten vollen Zugriff auf alle Module haben"
20.05.2026	Amann → Miller	Nachfrage, welche Schritte für die Modul-Aktivierung nötig sind (neue Installation? Reaktivierung? automatische Freischaltung?)

> Die Original-E-Mails sind separat archiviert (Mailbox `Keanu.Amann@illwerkevkw.at`, Betreff "Anfrage zur Nutzung einer Baramundi-Lösung für mein Abschlussprojekt").

9. Fortwährende Problematiken

PROBLEM 1: BARAMUNDI-SERVICE STARTET NACH UPDATE NICHT

Symptom: Die Baramundi Database Manager Suite löst nach einem Update einen Service-Neustart aus, der vom OS blockiert wird.



Lösung: Server-Reboot.

PROBLEM 2: BSERVER STARTET NACH ERSTINSTALLATION NICHT (SQL-LOGIN-FEHLER)

System: `winsrvbara1` (Windows Server 2022 Standard Evaluation, Build `10.0.20348`)
Software: bMS 2025 R2 (`25.2.130.0`)
Datum: 07.04.2026

Symptom

Nach der Installation lief der Dienst `bServer` nicht. Die anderen Baramundi-Dienste (`bLicenseManagement`, `bWebserver`) liefen.

Dienst	Status (vor Reparatur)	Starttyp
`bLicenseManagement`	Running	Automatic
`bServer`	**Stopped**	Automatic (Delayed)
`bWebserver`	Running	Automatic

Diagnose-Schritte

1. **Service-Konfiguration geprüft:**sc.exe qc bServerErgebnis: Binary unter `…\Management Server\bServer.exe`, Starttyp AUTO_START (DELAYED), Dienstkonto `LocalSystem`, Abhängigkeiten `RpcSs` und `LanmanServer`.

2. **Windows Event Log geprüft:** MSI-Installation am 07.04.2026 um 14:45 mit Exit-Code 0 — also keine fehlgeschlagene Installation.

3. **SQL Server geprüft:**Get-Service -Name "*SQL*"`SQL Server (SQLEXPRESS)` lief.

4. **Datenbank existiert:**sqlcmd -S ".\SQLEXPRESS" -Q "SELECT name FROM sys.databases"`baraSQL` vorhanden.

5. **bServer-Logdatei `bserver_2026-04-07.0.log` analysiert — entscheidend:**Erster Startversuch (~14:54):SqlException: A network-related or instance-specific error occurred while establishing a connection to SQL Server. The server was not found or was not accessible.
(provider: SQL Network Interfaces, error: 26 - Error Locating Server/Instance Specified)→ SQL Server war beim Dienststart noch nicht bereit (Race Condition).Zweiter Startversuch (~14:55):ProgramTerminationException: Cannot initialize database connection.
Internal message: Cannot open database "baraSQL" requested by the login.
The login failed. Login failed for user 'NT AUTHORITY\SYSTEM'.→ Das Dienstkonto `NT AUTHORITY\SYSTEM` hatte keinen Zugriff auf `baraSQL`.

Root Cause

Der `bServer`-Dienst läuft als `NT AUTHORITY\SYSTEM` (LocalSystem). Dieses Konto hatte zwar einen SQL Server Login, war aber:

1. Keiner Server-Rolle zugeordnet (nicht `sysadmin`, nicht `public`).
2. Nicht als Datenbank-User in `baraSQL` eingetragen.

Die Datenbank `baraSQL` gehörte dem User `T12\adminkan` (der die Installation durchgeführt hat). Der Database Manager konnte die DB mit dem aktuellen Windows-User öffnen, aber der `bServer`-Dienst lief als SYSTEM und hatte schlicht keine Berechtigung.

Lösung

1. **SQL-Berechtigung für `NT AUTHORITY\SYSTEM` einrichten:**USE baraSQL;
CREATE USER [NT AUTHORITY\SYSTEM] FOR LOGIN [NT AUTHORITY\SYSTEM];
ALTER ROLE db_owner ADD MEMBER [NT AUTHORITY\SYSTEM];

2. **Dienst starten:**Start-Service bServerErgebnis: Dienst kommt sauber hoch, alle Module initialisieren (Database Connection OK, BaraNet HTTP MOC auf `http://localhost:10081`, ASP.NET HtmlView auf `http://localhost:50313`, alle Service-Provider verbunden). Keine ERROR-Einträge mehr im Log.

Folgewarnung: "No current password found"

Nach dem erfolgreichen Start zeigte das Log in allen Service-Modulen (MOC, Apple, bConnect, NetworkDiscovery, Variables, WindowsCompliance, Networkscanner, SealedSoftware, OrgObjectService, NetworkHub, WindowsSoftwareInventory) wiederholt:

```
WARN   No current password found
       [Module=baramundi.bMS.DataAccess.DataContext]
```

Diagnose: Die Warnung bezieht sich auf das interne **StoragePassword** (`baramundi.bMS.DataAccess.StoragePassword`), das baramundi intern für die Verschlüsselung gespeicherter Daten verwendet (Domain-Credentials, Zertifikate, SSH-Keys etc.). Die `StoredEncryptedValue`-Tabelle enthielt bereits 31 verschlüsselte Einträge (DomainCredentialEncryptionPassword, CertificateAuthority, ConnectorCertificate, …). Die Domain `t12.lan` war korrekt mit den Konten `baraadmin`, `barainst` und `baranet` (verschlüsselte Passwörter vorhanden) konfiguriert.

Versuchte Maßnahme (führte nicht zum Verschwinden der Warnung):

1. `bServer`-Prozess gestoppt (PID 2744 via `Stop-Process -Force`, da START_PENDING keinen normalen Stop erlaubte).
2. Security-Settings zurückgesetzt:DBMgrCmd.exe -Action:ResetSecuritySettingsAusgabe: `Security settings successfully resetted.`
3. `bServer` neu gestartet: `sc.exe start bServer`.

Tatsächliches Verhalten: Die Warnung blieb auch nach dem Reset bestehen. Das ist bei einer **Erstinstallation erwartetes Verhalten** das StoragePassword wird automatisch beim **ersten Login über das baramundi Management Center (bMC)** gesetzt. Nach dem ersten erfolgreichen bMC-Login verschwindet die Warnung dauerhaft.

Status nach der Reparatur

Dienst	**Status**	**Starttyp**
`bLicenseManagement`	Running	Automatic
`bServer`	**Running**	Automatic (Delayed)
`bWebserver`	Running	Automatic

Lessons Learned / Vorbeugung

- Der Race-Condition-Anteil (Fehler 1) wird durch den Delayed-Start des `bServer` und das Service-Start-Hardening aus §5.3 (SQL Browser auf Automatic, `ServicesPipeTimeout = 120000`) entschärft.
- Falls der Dienst nach einem Reboot trotzdem nicht hochkommt: manuell `sc.exe start bServer` reicht in der Regel.
- Für künftige Baramundi-Installationen: die `NT AUTHORITY\SYSTEM`-Berechtigung auf die bMS-Datenbank ist Pflicht, nicht optional. Direkt nach `CREATE DATABASE baraSQL` mit anlegen.

PROBLEM 3: BARAMUNDI-MODULE NICHT VOLLSTÄNDIG SICHTBAR (OFFEN)

System: `winsrvbara1` (Windows Server 2022, bMS 2025 R2)

Stand: 27.05.2026 — **offen**, kommt mit in die Doku als bekannter Ausstand

Symptom

Nach erfolgreicher Lizenzaktivierung (06.05.2026, Ticket-Nr. siehe §8) sind in der bMS-Oberfläche nur die Module **Defense Control** und **Configuration/PXE** sichtbar. Das Anlegen von Jobs (Softwareverteilung, automatisierte Installationen, etc.) ist nicht möglich — die für das Abschlussprojekt zentral geplanten Szenarien lassen sich damit aktuell nicht umsetzen.

Bisheriger Verlauf

- 19.05.2026: Anfrage an `Agnes.Miller@baramundi.com` zur Prüfung des Lizenzumfangs.
- 20.05.2026: Hersteller hat die Lizenz bis 30.06.2026 verlängert mit dem Hinweis, der Zugriff auf alle Module sollte damit wieder funktionieren.
- 20.05.2026: Rückfrage von uns: ob für die Modul-Aktivierung eine Neuinstallation, eine erneute Aktivierung oder eine automatische Freischaltung nötig ist. **Antwort steht zum Stand 27.05.2026 noch aus.**
- 27.05.2026: Module nach wie vor nicht alle sichtbar; eine erneute Aktivierung mit demselben Ticket sowie ein Neustart der Baramundi-Dienste (`bServer`, `bLicenseManagement`, `bWebserver`) haben am Bild nichts geändert.

Vermutete Ursachen

Reihenfolge nach Wahrscheinlichkeit:

1. Die Lizenz ist serverseitig (beim Hersteller) noch nicht auf den erweiterten Modul-Umfang umgestellt die Verlängerung bis 30.06.2026 hat zwar die Laufzeit, aber nicht den Funktionsumfang berührt.

2. Eine erneute Aktivierung mit der Ticket-Nummer `8a3a7265-…` ist nötig, damit der `bLicenseManagement`-Dienst den neuen Modul-Stand vom Lizenzserver zieht.

3. Lokales Lizenz-Cache-File hält den alten Modul-Stand fest und müsste manuell zurückgesetzt werden (Hinweis dazu vom Hersteller nicht erhalten).

Stand: 2026-05-27.

10. Lessons Learned

Das Projekt hat nicht nur eine lauffähige Infrastruktur hervorgebracht, sondern auch eine Reihe von Erkenntnissen, die wir für künftige Projekte mitnehmen. Wir fassen sie hier bewusst offen zusammen, sowohl das, was gut funktioniert hat, als auch das, was uns Zeit und Nerven gekostet hat.

Was gut funktioniert hat

Am meisten gefreut hat uns, wie sauber sich unser selbst entwickelter Messenger Rede in die geschaffene Infrastruktur eingefügt hat. Wir haben von Anfang an auf eine Struktur gesetzt, die sich einfach erweitern und befüllen lässt, und genau das hat sich ausgezahlt: Den Rede-Relay-Server in DMZ, Firewall-Policies, VIP und WireGuard-Sync einzubinden, ging reibungslos und ohne dass wir bestehende Teile umbauen mussten. Das bestätigt, dass sich der zusätzliche Aufwand für ein durchdachtes, erweiterbares Grundgerüst am Anfang später mehrfach auszahlt.

Auch die Firewall und das SSL-VPN waren schneller eingerichtet als erwartet. Die FortiGate-Konfiguration ging zügig von der Hand; der einzige nennenswerte Stolperstein war das anfangs aktive Split-Tunneling, das VPN-Clients den Zugriff auf interne Ressourcen verwehrte. Nachdem wir Split-Tunneling komplett deaktiviert hatten (gesamter Client-Traffic durch den Tunnel), war das Problem gelöst, eine kleine Lektion darüber, wie stark eine einzelne Routing-Einstellung das Gesamtverhalten bestimmt.

Was schwierig war

Rede war das mit Abstand anspruchsvollste Teilstück. Die Kryptografie allein war komplex genug, aber der eigentliche Brocken war die Kommunikationsschicht: Text- und vor allem Sprachübertragung zuverlässig, verschlüsselt und über verschiedene Transportwege hinweg zum Laufen zu bringen, war ungleich mehr Arbeit als zunächst gedacht. Die vielen iterativen Fixes in der Client-Versionierung spiegeln diesen Aufwand wider. Die Lehre daraus: Bei selbst entwickelten Echtzeit-Kommunikationssystemen liegt der Aufwand weniger in der reinen Verschlüsselung als in den unzähligen Detailfällen der Synchronisation und des Transports.

Der größte vermeidbare Zeitfresser kam von außen: die Baramundi-Lizenz. Trotz erfolgreicher Aktivierung blieben die für das Projekt zentral geplanten Module nicht vollständig nutzbar und die Klärung mit dem Hersteller zog sich über Wochen hin, ohne dass das Problem bis zum Doku-Stand gelöst war (siehe §9, Problem 3). Eine reine Laufzeitverlängerung der Lizenz änderte am Funktionsumfang nichts.

Daraus ziehen wir eine klare Konsequenz für die Zukunft: Wir würden möglichst auf Open-Source-Lösungen setzen und uns nicht erneut in die Abhängigkeit eines Lizenzgebers begeben, dessen Freischaltungsprozesse und Reaktionszeiten wir nicht kontrollieren können.

Ähnliches gilt für die zweckentfremdete Hardware: Dass unsere Proxmox-Hosts in Wirklichkeit ausrangierte Kemp-Loadbalancer-Appliances sind, hat uns durchgehend zu Sorgfalt bei Monitoring und Backups gezwungen (siehe §2). Rückblickend war das eine gute Schule, es hat uns gelehrt, Ausfallrisiken von vornherein einzuplanen, statt sie zu ignorieren.

Eine wiederkehrende technische Lektion war zudem, dass eine Erstinstallation selten „einfach läuft": Beim Baramundi-bServer mussten wir erst über die Log-Analyse herausfinden, dass das Dienstkonto NT AUTHORITY\SYSTEM explizit Datenbankrechte braucht und der Dienst durch eine Race-Condition zu früh startete (siehe §9, Problem 2). Solche Abhängigkeiten stehen in keiner Anleitung, sie zu finden, gehört zur eigentlichen Arbeit.

Zusammenarbeit im Team

Die Aufteilung im Team hat von Anfang an gut funktioniert. Wir haben die Aufgaben früh nach unseren jeweiligen Stärken aufgeteilt, uns dabei aber durchgehend abgestimmt und Wissen ausgetauscht, statt in getrennten Silos zu arbeiten. Das hat verhindert, dass Wissen zu sehr auf eine Person konzentriert war, und sorgte dafür, dass beide jederzeit den Überblick über das Gesamtsystem behielten.

Fazit

Unterm Strich hat das Projekt gezeigt, dass eine gut durchdachte Grundstruktur, klare Aufgabenteilung und durchgehende Kommunikation die wichtigsten Erfolgsfaktoren waren. Die größten Reibungspunkte entstanden dort, wo wir von externen Faktoren abhängig waren, sei es ein Lizenzgeber oder die Grenzen geschenkter Hardware. Genau das nehmen wir als zentrale Lehre mit: So viel wie möglich in der eigenen Hand behalten.