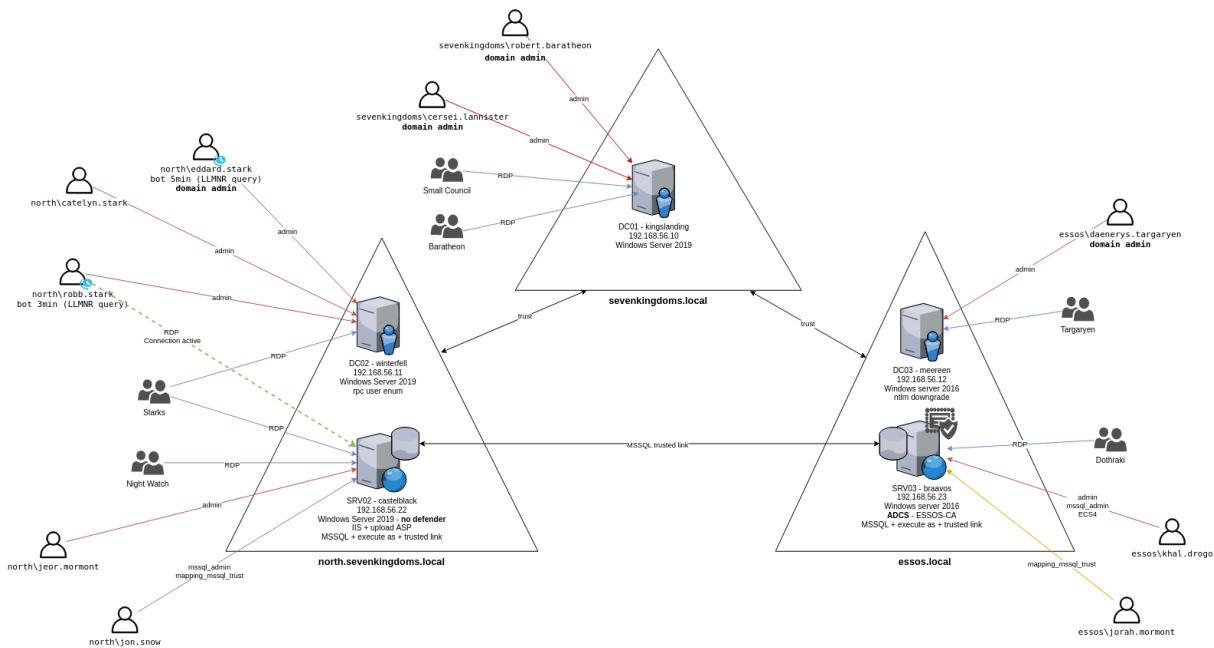


Promox GOAD (Game of Active Directory) Lab Setup

 koller.ninja/proxmox-goad-lab

Security Things

February 19, 2024



Intro

Recently, I came across this article from [@M4yFly](#) - [Orange-Cyberdefense/GOAD: game of active directory](#). I'm always on the lookout for labs for Infosec practice and the installation options seemed great.

- Installation depend of the provider you use, please follow the appropriate guide :
 - [Install with Virtualbox](#)
 - [Install with VmWare](#)
 - [Install with Proxmox](#)
 - [Install with Azure](#)

install section

I have an older desktop form factor server and was running ESXi on it. I had previously had VMware workstation on it as well and generally try to stay away from VirtualBox . I've been frustrated with ESXi as the version I am able to run can't emulate TPMs for Windows 11 and you have to buy the \$500 license if you want backup features, etc. Additionally, just announced, the free version of ESXi is sunseting.

I decided to take the leap and re-image/configure the server with Proxmox. That was a trip in and of itself and maybe I'll write about that later.

So I started with [@M4yFly's](#) Proxmox setup here: [GOAD on proxmox - Part1 - Proxmox and pfsense](#). All credit goes to him, this is just some stuff that I came across that I had to edit to get it working for me. It may not apply to your setup, but if it helps someone, great.

Warning! - The whole original setup is LONG - broken up into 5 parts just for the setup. This was fine for me as I was learning Proxmox (and Pfsense) as I went along. Also, the original writeup uses a hosted Proxmox service and my server was in my house. I've edited things as needed to keep things local. This includes the Pfsense setup and the VPN connection. May seem strange to fire up the VPN to connect to the lab when it's all local, but I kinda liked that idea from a segmentation perspective.

I'll not re-post all the steps here, just the parts that are either different from my setup or that I had to edit to get working for me. He [has published](#) an update to the repo recently which I have not had a chance to test yet, so your mileage may vary.

Part 1: Proxmox and pfsense

I ignored the cloud setup for Proxmox as mine is local. Whether it was me being new to Pfsense or not following directions, the Pfsense section had me rebuilding a few times before I got it right. The Pfsense setup pictures seemed a little off from what I was seeing. Here is a screen shot from my setup if it helps. You need not worry about the VPN part at this stage.

```
*** Welcome to pfSense 2.7.2-RELEASE (amd64) on pfSense ***

WAN (wan)      -> vtnet0      -> v4: 10.0.0.2/30
LAN (lan)      -> vtnet1      -> v4: 192.168.2.2/24
OPT1 (opt1)    -> vtnet2      ->
VLAN10 (opt2)  -> vtnet2.10   -> v4: 192.168.10.1/24
VLAN20 (opt3)  -> vtnet2.20   -> v4: 192.168.20.1/24
GOADVPN (opt4) -> ovpn1       -> v4: 10.10.10.1/32
```

The ssh config [section](#) is the IP of my local Proxmox server instead of the public IP. Pay careful attention to the instructions on setting up the firewall rules. The rest was all good for me.

Part 2: Templating with Packer

I use my Proxmox for various things on my network, so it's not all used for the lab. Because of that I wanted to use ZFS for the file system for various drive pools. It seems that because of this decision, I had to use the `vm_disk_format` of "raw" instead of the default "qcow2" format. I know it comes with trade-offs, but that is what worked for me. It will not build without changing that.

So I edited the files below as such.

`/GOAD/packer/proxmox/windows_server2016_proxmox_cloudinit.pkvars.hcl`

```

winrm_username = "vagrant"
winrm_password = "vagrant"
vm_name = "WinServer2016x64-cloudinit-qcow2"
template_description = "Windows Server 2016 64-bit - build 14393 - template built
with Packer - cloudinit - "
iso_file = "local:iso/windows_server_2016_14393.0_eval_x64.iso"
autounattend_iso = "./iso/Autounattend_winserver2016_cloudinit.iso"
autounattend_checksum =
"sha256:faa1b05c7cd8b8545698378ef34efbbba3b71ba754012c7e0f642741127326c8"
vm_cpu_cores = "2"
vm_memory = "4096"
vm_disk_size = "40G"
vm_sockets = "1"
os = "win10"
vm_disk_format = "raw"

```

/GOAD/packer/proxmox/windows_server2019_proxmox_cloudinit.pkvars.hcl

```

winrm_username      = "vagrant"
winrm_password      = "vagrant"
vm_name             = "WinServer2019x64-cloudinit-raw"
template_description = "Windows Server 2019 64-bit - build 17763.737.190906-2324
- template built with Packer - cloudinit - "
iso_file            = "local:iso/windows_server2019_x64FREE_en-us.iso"
autounattend_iso    = "./iso/Autounattend_winserver2019_cloudinit.iso"
autounattend_checksum =
"sha256:65f77989f237d8921478315324e6570bd500369aa6783e2fe9eac7aaa729a899"
vm_cpu_cores        = "2"
vm_memory            = "4096"
vm_disk_size        = "40G"
vm_sockets          = "1"
os                   = "win10"
vm_disk_format      = "raw"

```

I also do not have a "proxmox_pool" in my setup, so I leave that blank for this file:

/GOAD/packer/proxmox/config.auto.pkrvars.hcl

```

proxmox_url          = "https://192.168.1.77:8006/api2/json"
proxmox_username     = "infra_as_code@pve"
proxmox_password     = "yourpasswordhere"
proxmox_skip_tls_verify = "true"
proxmox_node         = "proxmox"
proxmox_pool         = ""
proxmox_storage      = "Data"

```

Part 3: Providing with Terraform

When trying to build the VMs out with Terraform, I ran into an issue where it would not build. Threw some errors and ends up like this:

Error: The terraform-provider-proxmox_v2.9.14 plugin crashed!

This is always indicative of a bug within the plugin. It would be immensely helpful if you could report the crash with the plugin's maintainers so that it can be fixed. The output above should help diagnose the issue.

The solution I came across was using a different Proxmox provider. There were some threads about the default one not being updated, but for whatever reason it did not work for me.

I changed my main.tf file as below to add the different provider (version numbers from late 2023). /GOAD/ad/GOAD/providers/proxmox/terraform/main.tf

```
terraform {
  required_providers {
    proxmox = {
      source  = "telmate/proxmox"
      version = ">= 2.9.14"
    }
    proxmox = {
      source  = "thegameprofi/proxmox"
      version = ">= 2.9.15"
    }
  }
}

provider "proxmox" {
  pm_api_url = var.pm_api_url
  pm_user    = var.pm_user
  pm_password = var.pm_password
  pm_debug   = true
  pm_tls_insecure = true
  pm_parallel = 3
  pm_timeout  = 2400

  pm_log_enable = true
  pm_log_file   = "terraform-plugin-proxmox.log"
  pm_log_levels = {
    _default = "debug"
    _capturelog = ""
  }
}
```

Again, a few edits to the variables.tf file as well to indicate my local server and no Proxmox pool. /GOAD/ad/GOAD/providers/proxmox/terraform/variables.tf

```

variable "pm_api_url" {
    default = "https://192.168.1.77:8006/api2/json"
}

variable "pm_user" {
    default = "infra_as_code@pve"
}

variable "pm_password" {
    default = "<yourpasseordhere>"
}

variable "pm_node" {
    default = "proxmox"
}

variable "pm_pool" {
    default = ""
}

variable "pm_full_clone" {
    default = false
}

```

Then everything built out fine.

Part 4: Provisioning with Ansible

I needed to update the Ansible inventory file by adding “Ethernet 2” in the adapter section as shown below. Otherwise it errored out for me.

/GOAD/ad/GOAD/providers/proxmox/inventory

```

[all:vars]
; domain_name : folder inside ad/
domain_name=GOAD

force_dns_server=yes
dns_server=192.168.10.1

two_adapters=no
; adapter created by vagrant and virtualbox (comment if you use vmware)
nat_adapter=Ethernet 2
domain_adapter=Ethernet 2

; adapter created by vagrant and vmware (uncomment if you use vmware)
; nat_adapter=Ethernet0
; domain_adapter=Ethernet1

```

Part 5: VPN Access with OpenVPN

After following the VPN setup, I was not able to connect to the firewall from my local Mac laptop using the OpenVPN client. The key for me was enabling the interface for the VPN in Pfsense. I'll show some finished screen shots in hopes that may help someone.

Interfaces / [goadvpn \(ovpns1\)](#)

General Configuration

Enable

☒ Enable interface

Description

Enter a description (name) for the interface here.

IPv4/IPv6 Configuration

This interface type does not support manual address configuration on this page.

MTU

If this field is blank, the adapter's default MTU will be used. This is typically 1500 bytes but can vary in some circumstances.

MSS

If a value is entered in this field, then MSS clamping for TCP connections to the value entered above minus 40 for IPv4 (TCP/IPv4 header size) and minus 60 for IPv6 (TCP/IPv6 header size) will be in effect.

Reserved Networks

Block private networks and loopback addresses

☐
Blocks traffic from IP addresses that are reserved for private networks per RFC 1918 (10/8, 172.16/12, 192.168/16) and unique local addresses per RFC 4193 (fc00::/7) as well as loopback addresses (127/8). This option should generally be turned on, unless this network interface resides in such a private address space, too.

Block bogon networks

☐
Blocks traffic from reserved IP addresses (but not RFC 1918) or not yet assigned by IANA. Bogons are prefixes that should never appear in the Internet routing table, and so should not appear as the source address in any packets received. This option should only be used on external interfaces (WANs), it is not necessary on local interfaces and it can potentially block required local traffic. Note: The update frequency can be changed under System > Advanced, Firewall & NAT settings.

Interfaces / [Interface Assignments](#)

Interface Assignments

Interface Groups

Wireless

VLANs

QinQs

PPPs

GREs

GIFs

Bridges


































LAGGs

| Interface | Network port |
|-----------|---|
| WAN | <input type="text" value="vtnet0 (bc:24:11:da:8a:e6)"/> |
| LAN | <input type="text" value="vtnet1 (bc:24:11:1a:b8:51)"/> Delete |
| OPT1 | <input type="text" value="vtnet2 (bc:24:11:a3:2b:62)"/> Delete |
| VLAN10 | <input type="text" value="VLAN 10 on vtnet2 - opt1 (VLAN10)"/> Delete |
| VLAN20 | <input type="text" value="VLAN 20 on vtnet2 - opt1 (VLAN20)"/> Delete |
| goadvpn | <input type="text" value="ovpns1 (goad-vpn)"/> Delete |

[Save](#)



















Floating WAN LAN OPT1 VLAN10 VLAN20 GOADVPN OpenVPN

Rules (Drag to Change Order)

|  | States | Protocol | Source | Port | Destination | Port | Gateway | Queue | Schedule | Description | Actions |
|---|---|-------------|-------------------------------------|------|----------------|--------------|---------|-------|----------|---|---|
|  | 0/0 B | * | Reserved Not assigned by IANA | * | * | * | * | * | | Block bogus networks |  |
| IN | | | | | | | | | | |  |
|  |  0/188 KiB | IPv4 UDP | * | * | WAN address | 2137 | * | none | | goad-OpenVPN |      |
|  |  0/0 B | IPv4 TCP | 10.0.0.1 | * | 192.168.2.3 | 22 (SSH) | * | none | | Allow ssh tunnel provisioning |      |
|  |  0/0 B | IPv4 TCP | 10.0.0.1 | * | LAN address | 80 (HTTP) | * | none | | Allow ssh tunnel for pfsense http access |      |
| Default Deny | | | | | | | | | | |  |
|  |  0/1.26 MiB | IPv4 TCP | * | * | * | * | * | none | | |      |




 Add  Add  Delete  Toggle  Copy  Save  SeparatorFloating WAN LAN OPT1 VLAN10 VLAN20 GOADVPN OpenVPN

Rules (Drag to Change Order)

|  | States | Protocol | Source | Port | Destination | Port | Gateway | Queue | Schedule | Description | Actions |
|---|---|----------|--------|------|----------------|------|---------|-------|----------|--------------------|---|
| IN | | | | | | | | | | |  |
|  |  0/115 KiB | IPv4+6 * | * | * | VLAN10 subnets | * | * | none | | allow vlan1 access |       |
| Default Deny | | | | | | | | | | |  |
|  |  0/0 B | IPv4 * | * | * | * | * | * | none | | block all |      |

 Add  Add  Delete  Toggle  Copy  Save  SeparatorServers Clients Client Specific Overrides Wizards Client Export

OpenVPN Servers

| Interface | Protocol / Port | Tunnel Network | Mode / Crypto | Description | Actions |
|-----------|----------------------|----------------|--|-------------|---|
| WAN | UDP4 / 2137 (TUN) | 10.10.10.0/24 | Mode: Remote Access (SSL/TLS + User Auth) Data Ciphers: AES-256-GCM, AES-128-GCM, CHACHA20-POLY1305, AES-256-CBC Digest: SHA256 D-H Params: 2048 bits | goad-vpn |    |

 Add

OpenVPN / Client Export Utility

Server
Client
Client Specific Overrides
Wizards
Client Export

OpenVPN Server

Remote Access Server

goad-vpn UDP4:2137

Client Connection Behavior

Host Name Resolution

Other

Host Name

192.168.1.77

Enter the hostname or IP address the client will use to connect to this server.

Verify Server CN

Automatic - Use verify-x509-name where possible

Optionally verify the server certificate Common Name (CN) when the client connects.

Block Outside DNS

☐ Block access to DNS servers except across OpenVPN while connected, forcing clients to use only VPN DNS servers.

Requires Windows 10 and OpenVPN 2.3.9 or later. Only Windows 10 is prone to DNS leakage in this way, other clients will ignore the option as they are not affected.

Legacy Client

☐ Do not include OpenVPN 2.5 and later settings in the client configuration.

When using an older client (OpenVPN 2.4.x), check this option to prevent the exporter from placing known-incompatible settings into the client configuration.

Silent Installer

☐ Create Windows installer for unattended deploy.

Create a silent Windows installer for unattended deploy; installer must be run with elevated permissions. Since this installer is not signed, you may need special software to deploy it correctly.

Bind Mode

Do not bind to the local port

If OpenVPN client binds to the default OpenVPN port (1194), two clients may not run concurrently.

As a bonus earlier on, he gives a script for creating snapshots of the VMs. In addition to this, I wanted a way to spin up/down the lab VMs when I wasn't using them. Update as needed.

Shutdown:

```
vms=("GOAD-SRV03" "GOAD-SRV02" "GOAD-DC01" "GOAD-DC02" "GOAD-DC03" "pfsense")
COMMENT="Shutdown"
# Loop over the array
for vm in "${vms[@]}"
do
    echo "[+] Shutting down $vm"
    id=$(qm list | grep $vm | awk '{print $1}')
    echo "[+] VM id is : $id"
    qm shutdown "$id"
done
```

Startup:

```
vms=("GOAD-SRV03" "GOAD-SRV02" "GOAD-DC01" "GOAD-DC02" "GOAD-DC03" "pfsense")
COMMENT="Start"
# Loop over the array
for vm in "${vms[@]}"
do
    echo "[+] Starting $vm"
    id=$(qm list | grep $vm | awk '{print $1}')
    echo "[+] VM id is : $id"
    qm start "$id"
done
```


Have fun.