PROGETTO DI ARCHITETTURA DI RETE E PREVENTIVO TECNICO-ECONOMICO PER L'IMPLEMENTAZIONE DI INFRASTRUTTURE DI SICUREZZA PERIMETRALE E INTERNA

Proposta Tecnica in Risposta all'Esigenza della Compagnia Theta (Theta Co.)

SOMMARIO ESECUTIVO

Il presente documento rappresenta la Technical Proposal e l'Offerta tecnico-economica di **VT Firewall S.r.l.** per la progettazione e l'implementazione di una nuova architettura di rete resiliente per conto della Compagnia Theta.

L'obiettivo di questo progetto è fornire alla Compagnia Theta una moderna architettura IT scalabile e soprattutto **sicura**, in grado di supportare complessivamente **120 postazioni** utente distribuite su **sei piani** dell'immobile

Visione del Progetto

La progettazione è stata condotta tenendo in alta considerazione la necessità di segregazione e protezione dei dati in base alla loro criticità, riflettendo la suddivisione funzionale dei reparti all'interno dell'edificio:

- **Piani a Rischio Elevato/Sensibile:** Il Piano Terra (Reparto IT, Risorse Umane), il Secondo Piano (Amministrazione, Finanza e Controllo) e il Terzo Piano (Centro R&S) sono stati identificati come gestori di dati altamente sensibili e proprietà intellettuali di grande valore.
- **Architettura di Sicurezza:** La soluzione proposta integra un **Firewall perimetrale**, una **Zona Demilitarizzata (DMZ)** per l'hosting del Web Server (DVWA), e un sistema a triplo strato di **IDS/IPS** (Intrusion Detection/Prevention System) per un monitoraggio approfondito del traffico interno.

Il report illustrerà in dettaglio la topologia di rete, il preventivo dei costi hardware e software, e documenterà i test di sicurezza eseguiti per garantirne l'efficacia operativa.

Panorama Requisiti e Proposta di Infrastruttura di Rete

1.1 Analisi dei Requisiti Funzionali e Fisici

L'analisi dei requisiti di base conferma la necessità di implementare una solida infrastruttura di rete dimensionata per l'attuale struttura operativa del cliente:

- **Struttura e Host:** Il progetto si sviluppa su un edificio di **6 piani**. La rete deve essere in grado di supportare stabilmente **120 dispositivi finali** (computer), con una previsione di 20 postazioni per piano.
- **Componenti di Sicurezza:** Oltre ai dispositivi standard, l'architettura deve integrare un **Firewall perimetrale** , **tre IDS/IPS** per il monitoraggio interno e un **NAS** per l'archiviazione centralizzata dei dati aziendali.

1.2 Suddivisione dei Reparti e Sensibilità dei Dati

La suddivisione fisica dei reparti su sei piani non è solo logistica, ma impone una chiara strategia di **segmentazione logica (VLAN)** per isolare il traffico e applicare politiche di sicurezza granulari. I dati gestiti da ciascun piano presentano livelli di sensibilità differenti, che giustificano l'isolamento:

- **Piano Terra: Reception, Risorse Umane (HR) e Reparto IT**.
  - **Sensibilità: Elevata**. L'HR gestisce dati personali riservati e contratti. Il Reparto IT è il punto di controllo della rete (CED), rendendo questo piano strategico e necessitando di un isolamento rigoroso.
  - **Giustificazione Segmentazione:** Isolamento del traffico IT per manutenzione e controllo separato dell'HR (dati PII).
- **Primo Piano: Ufficio Commerciale e Marketing**.
  - **Sensibilità: Media/Alta**. Gestione di liste clienti, strategie di vendita e dati di promozione.
- **Secondo Piano: Amministrazione, Finanza e Controllo**.
  - **Sensibilità: Critica**. Questo è il cuore finanziario dell'azienda; i dati sono altamente sensibili (conti, pagamenti, strategie finanziarie) e richiedono la massima protezione contro accessi non autorizzati.
  - **Giustificazione Segmentazione:** Protezione assoluta del traffico finanziario tramite politiche Firewall/ACL dedicate, anche a livello inter-VLAN.
- **Terzo Piano: Centro Ricerca e Sviluppo (R&S)**.
  - **Sensibilità: Critica (Proprietà Intellettuale - IP)**. Gestione di idee, prototipi e tecnologie future. La perdita o il furto di questi dati comprometterebbe il vantaggio competitivo della Compagnia Theta.
  - **Giustificazione Segmentazione:** Isolamento totale per prevenire l'esfiltrazione di IP e restrizioni severe sull'accesso al NAS e a Internet.
- **Quarto Piano: Ingegneria di Prodotto e Sviluppo Software**.
  - **Sensibilità: Alta**. Trasformazione dell'IP in codice e prodotti concreti.
- **Quinto Piano: Divisione Operativa e Supporto Clienti**.
  - **Sensibilità: Media**. Gestione logistica e assistenza post-vendita (dati di soddisfazione e fidelizzazione).
- **Sesto Piano (Attico): Direzione Generale e Uffici Legali**.
  - **Sensibilità: Critica (Centro Decisionale)**. Dati strategici, legali e di top management (CEO, CFO, ecc.).
  - **Giustificazione Segmentazione:** Massima priorità di accesso e isolamento per garantire riservatezza e integrità del processo decisionale.

2 Preventivo Economico-Tecnico

Il presente preventivo **include i materiali e i dispositivi necessari** per l'implementazione della rete,

nonché i relativi costi tecnici.

Secondo le specifiche da Voi richieste

La proposta economica include

- **Cavi Ethernet cat 6;**
- **N. 6 switch per piano;**
- **N. 1 server NAS per archiviazione centralizzata;**
- **N. 1 web server esterno (posizionato in DMZ);**
- **N. 1 router aziendale;**
- **N. 3 dispositivi IDS/IPS per il monitoraggio e la prevenzione delle intrusioni;**
- **N. 1 firewall perimetrale.**

Sono inoltre compresi nel preventivo:

- Il **lavoro dei tecnici** specializzati per il cablaggio e l'installazione dei dispositivi di rete;
- Il **supporto operativo** per il collegamento e la configurazione dei terminali aziendali;
- **Le verifiche tecniche**, comprensive della risoluzione delle problematiche relative alla configurazione del subnetting;
- **L'intervento del team** di sviluppo per l'integrazione dei software richiesti all'interno della rete.

è inoltre riportata **una stima dettagliata dei costi** e delle tempistiche previste per lo sviluppo dei tre applicativi

software richiesti a supporto dell'infrastruttura:

- Verifica dei Verbi http:
  Software per inviare richieste HTTP (GET, POST, PUT, DELETE) al web server e verificare le risposte.
- Scansione delle Porte:
  Software per eseguire una scansione delle porte sui dispositivi di rete, verificando la sicurezza e l'accessibilità delle varie porte di comunicazione.
- Socket di Rete: Programma che catturi il Socket di rete.

Di seguito il preventivo dettagliato:

COSTI MATERIALI: (tutti i prezzi sono compresi di iva)				
Voce	Dettagli	Quantità	Costo unitario (€)	Totale (€)
Cavi Ethernet	Cavo cat 6a	2500m	1,9/m	4.750,00 €
Web server esterno	Server di fascia media	1	1.600,00 €	1.600,00 €
Firewall perimetrale	(Fortigate/FortiCare/FortiGuard)	1	1.192,380 €	1.192,38 €
Router Aziendale	Mikrotik RB4011IGS+RM	1	197,72 €	197,72 €
Switches	NebulaFlex Cloud	6	299,00 €	1.794,00 €
Server NAS	LincPlus LincStation N2	1	461,95 €	461,95 €
Dispositivi IDS/IPS	FortiGate 60F	3	1.300,00 €	3.900,00 €
Computer aziendali (compresi di Tastiera, Mouse, schermo	CPU: Intel i3 14100F 4 P-core +0 E-core RAM 8Gb SSD 256Gb	103	500,00 €	51.500,00 €
Computer amministrazione (compresi di Tastiera, Mouse, Schermo)	CPU: Intel i5-14600K 6 P-core + 8 E-core RAM: 16Gb SSD: 512Gb	10	700,00 €	7.000,00 €
Computer reparto IT (compresi di Tastiera, Mouse, Schermo)	CPU: Intel i7 14700k 8 P-core + 12 E-core RAM: 32Gb SSD: 1T	7	1.100,00 €	7.700,00 €
TOTALE MATERIALE				80.096,05

COSTI TECNICI E DI MANODOPERA: (tutti i prezzi sono compresi di iva)				
Voce	Dettagli	Giorni lavorativi	Costo unitario (€)	Totale (€)
Tecnici installazione switch	Installazione e cablaggio di 6 switch (20 PC per piano)	1	6 Tecnici - €300,00/cad.	1.800,00 €
Tecnici configurazione rete	Configurazione IP, DHCP - Gateway e verifica funzionalità rete	1	6 Tecnici - €300,00/cad.	1.800,00 €
Tecnici IDS/IPS	Installazione e configurazione dei 3 dispositivi IDS/IPS	1	2 Tecnici - €350,00/cad.	700,00 €
Tecnico firewalls perimetrale	Installazione e configurazione firewall	1	1 Tecnico – € 320,00/cad.	320,00 €
Tecnico Web Server	Configurazione Web Server (Metasploitable)	1	1 Tecnico - € 280,00/cad.	280,00 €
Tecnico WAN	Configurazione uscita verso WAN esterna	1	1 Tecnico - € 280,00/cad.	280,00 €
TOTALE MATERIALE				5.180,00 €

COSTI TECNICI VERIFICHE EXTRA: (tutti i prezzi sono compresi di iva)				
Voce	Dettagli	Giorni lavorativi	Costo unitario (€)	Totale (€)
Verifica tecnica extra		5 ore	35/ora	175,00 €
Verifica hardware	1 Verifica	1	50,00/cad.	50,00 €
TOTALE MATERIALE				225,00 €

Sviluppo Software – Team di Programmazione (tutti i prezzi sono compresi di iva)					
Progetto/Strumento	Descrizione	Giorni lavorativi	N.Programmatori	Costo giornaliero (€)	Totale (€)
Verifica Verbi HTTP	Invio richieste HTTP (GET, POST, PUT, DELETE)	3	2 Programmatori	300,00/cad.	1.800,00 €
Scansione delle Porte	Scansione delle porte aperte sui dispositivi di rete	2	1 Programmatore	300,00/cad.	1.200,00 €
Cattura Socket di Rete	intercettare e analizzare Socket di rete	3	1 Programmatore	300,00/cad.	900,00 €
TOTALE MATERIALE					3.900,00 €

TESTING FASE E VALIDAZIONE SOFTWARE (tutti i prezzi sono compresi di iva)				
Fase	Giorni di Lavoro	Tecnici Coinvolti	Tariffa Giornaliera (€)	Totale (€)
Test & Debug	2	2 Programmatori	300,00 €	1.200,00 €
Validazione finale + Documentazione	1	1 Tecnico Senior	350,00 €	350,00 €
TOTALE MATERIALE				1.550,00 €

COSTI MATERIALI (tutti i prezzi sono da considerarsi IVA inclusa)				
Voce	Dettagli	Quantità	Costo unitario (€)	Totale (€)
Batteria NAS	Modulo 12 Batterie Aggiuntive per Gruppi di Continuità	1	1.223,79 €	1.223,79 €
Rack per server	Armadio per server Grande 26U Startech, 580 x 1265 x 550mm	1	1.002,30 €	1.002,30 €
Costo trasporto		1	450,00 €	450,00 €
Gruppo di continuita	APC SRTBKXLI (8000VA / 7200W	1	8.543,00 €	8.543,00 €
Antivirus per Computer	VT Antivirus	1	0,00 €	0,00 €
TOTALE MATERIALE				11.219,09 €

COSTI DI MANODOPERA:				
Voce	Dettagli	Giorni lavorativi	Costo unitario (€)	Totale (€)
Manodopera	Manodopera montaggio		40.000,00 €	40.000,00 €
TOTALE DI TUTTI I COSTI *Mate iali -Tecnici - Sviluppo Software e Testing Fase - Gestione*				142.170,14 €

Totali tempistiche Globali previste:

1. Preparazione e Installazione Infrastruttura di Rete (corrugati, canaline, cablaggio, switch, router, NAS, firewall, IDS/IPS):

Durata stimata: 5-7 giorni lavorativi

- **Attività coinvolte:** installazione canaline, posa cavi, montaggio e configurazione switch, router, firewall,NAS, IDS/IPS.
- **Personale coinvolto:** tecnici elettricisti, installatori di rete, sistemisti.

2. Configurazione e Test Rete Aziendale

Durata stimata: 3-4 giorni lavorativi

- **Attività:** assegnazione IP statici, Gateway, verifica DHCP, test di comunicazione interna, configurazione della WAN e del web server.

3. Sviluppo Software Personalizzato

Durata stimata: 8 giorni lavorativi (già indicata nei dettagli: 3 + 2 + 3 giorni)

Include sviluppo dei 3 tool software richiesti:

- Verifica Verbi HTTP
- Scansione Porte
- Cattura Socket di rete

4. Testing, Debug e Validazione Finale

Durata stimata: 2-3 giorni lavorativi

- Comprende test dei programmi, debug, documentazione e approvazione del funzionamento dell'intera infrastruttura software e hardware.

TOTALE TEMPISTICHE GLOBALI PREVISTE:

Circa 30 giorni lavorativi, salvo imprevisti (ritardo della fornitura dei materiali o Vostre richieste aggiuntive)

Clausole finali

Dal suddetto preventivo sono escluse tutti i lavori di edilizia quali: muratura impianti elettrici

Il presente preventivo costituisce parte integrante del contratto tra le parti, ai sensi dell'art. 1321 del Codice Civile, e sarà vincolante a partire dalla firma di accettazione da parte del committente.

È redatto su richiesta della Spett.le Theta Company, a seguito del pre-accordo già intercorso tra le parti.

Si conferma l'avvenuto versamento dell'anticipo concordato del 20%, necessario per l'avvio delle attività preliminari – cioè il seguente preventivo.

Al completamento della metà dei lavori, verrà corrisposto un ulteriore 50% dell'importo pattuito, in ottemperanza agli accordi prestabiliti.

Il saldo finale dovrà essere corrisposto entro e non oltre 60 giorni dalla conclusione dei lavori, previa emissione di regolare fattura.

Eventuali interventi aggiuntivi non previsti nel presente documento saranno oggetto di ulteriore preventivo.

Il presente documento ha validità 30 giorni dalla data di emissione, salvo diversi accordi scritti.

Trascorso tale termine, i prezzi potranno subire variazioni, salvo diversa indicazione scritta.

---

3. Schema di Rete e Topologia Logica

L'architettura adottata da **VT Firewall S.r.l.** è una topologia **Stella Estesa (Extended Star)**, con il Router centrale (ISR 4331) che funge da *Core Distribution*. Questa scelta è stata dettata dalla necessità di centralizzare l'intelligenza di routing e le politiche di sicurezza, garantendo al contempo:

- **Scalabilità:** Facile aggiunta di nuovi piani o reparti senza riconfigurare l'intera dorsale.
- **Gestibilità:** Un singolo punto di controllo per il routing inter-VLAN.
- **Resilienza:** Un guasto su uno Switch di piano isola il problema a quel livello, senza compromettere l'intera rete.

Il progetto è basato sul principio della **Difesa in Profondità (Defense in Depth)**, che applica controlli multipli in ogni punto critico: dai filtri del Firewall perimetrale alla segregazione interna tramite ACL.

3.1 Topologia Fisica

La rete adotta una topologia fisica a stella estesa. Questa architettura è evidente dagli schemi, dove più gruppi di dispositivi finali sono collegati a un dispositivo di rete centrale per ciascun segmento.

I punti chiave da descrivere sono:

- **Dispositivi Centrali**: Ogni segmento di rete converge verso un apparato centrale. Questi dispositivi agiscono come hub di comunicazione per tutti i dispositivi collegati.
- **Connessioni:** Tutti i dispositivi endpoint sono collegati direttamente al loro rispettivo switch/router centrale tramite cavi di rete.
- I**nterconnessione**: I vari dispositivi centrali sono a loro volta interconnessi, per consentire la comunicazione tra i diversi segmenti.

Questo tipo di topologia è stato scelto per la sua scalabilità e robustezza: un guasto a un singolo cavo o dispositivo endpoint non compromette la funzionalità del resto della rete.

3.2 Rappresentazione Grafica della Topologia

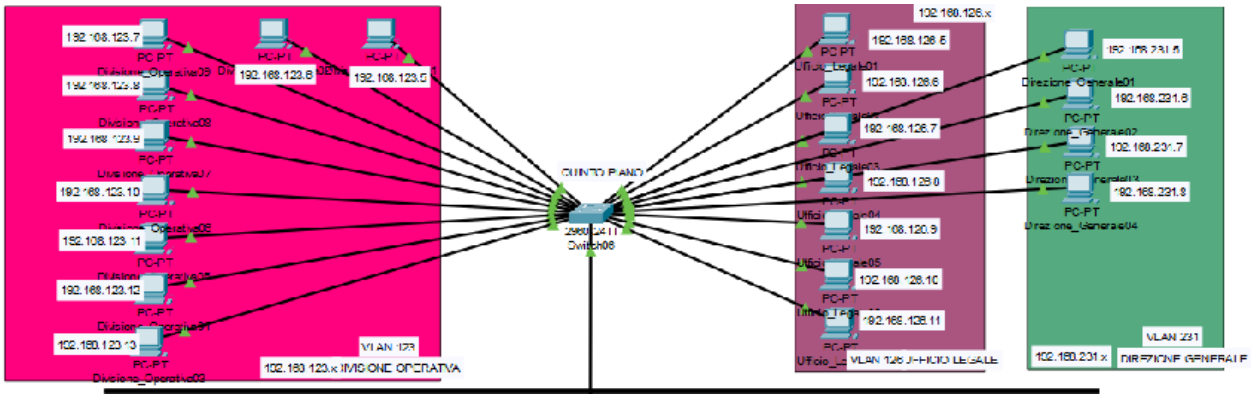
Lo schema qui presentato è il risultato finale della fase di progettazione e di validazione, in linea con i requisiti della Compagnia Theta. Per garantire l'affidabilità e la replicabilità dell'infrastruttura, la configurazione completa è stata simulata in un ambiente virtuale che rispecchia la topologia fisica dell'edificio

Quinto piano

Riservato al top management e al team legale

Reti e Componenti: 20 PC su Switch dedicato.

- VLAN123/Divisione Operativa:192.168.123.x/24
- VLAN128/Ufficio Legale: 192.168.128.x/24
- VLAN231/Direzione Generale: 192.168.231.x/24

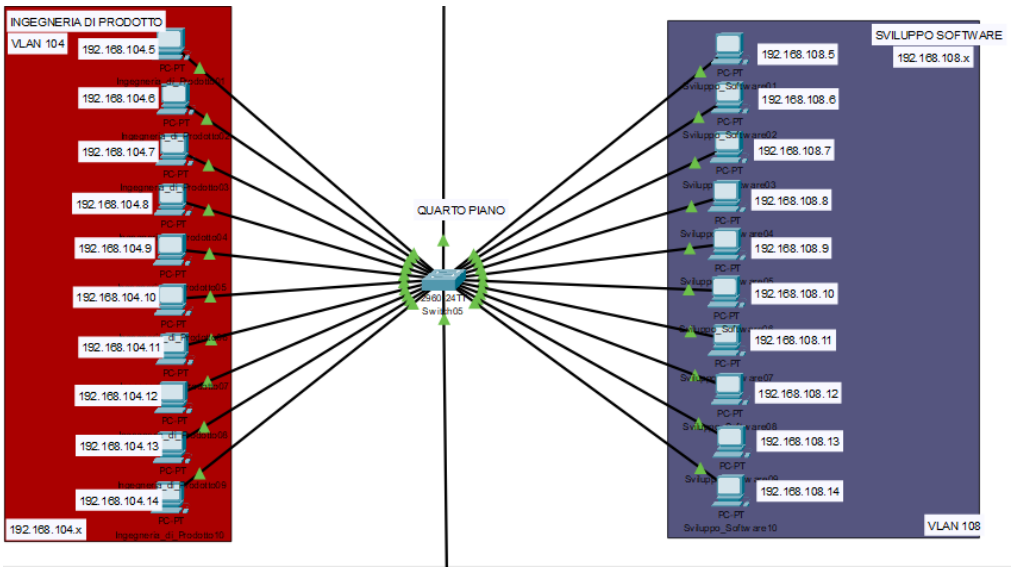


Quarto piano

Il team che concretizza le idee dell'R&S

Reti: 20 PC su Switch dedicato.

- **VLAN104/Ingegneria Di Prodotto:** 192.168.104.x/24
- **VLAN108/Sviluppo Software:** 192.168.108.x/24

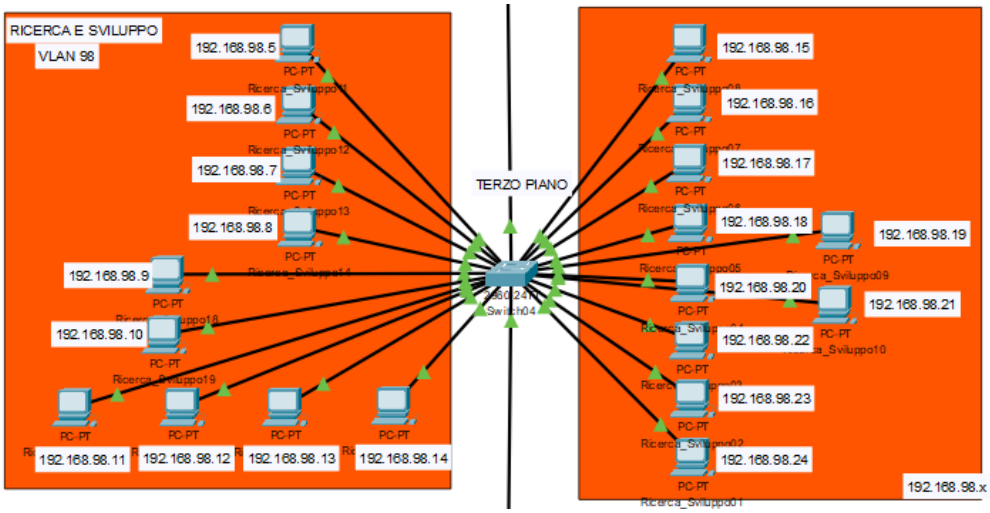


Terzo piano

Il reparto strategico per l'innovazione e la proprietà intellettuale (IP).

Reti e Componenti: 20 PC su Switch dedicato.

- **VLAN98/R&S:** 192.168.98.0/24

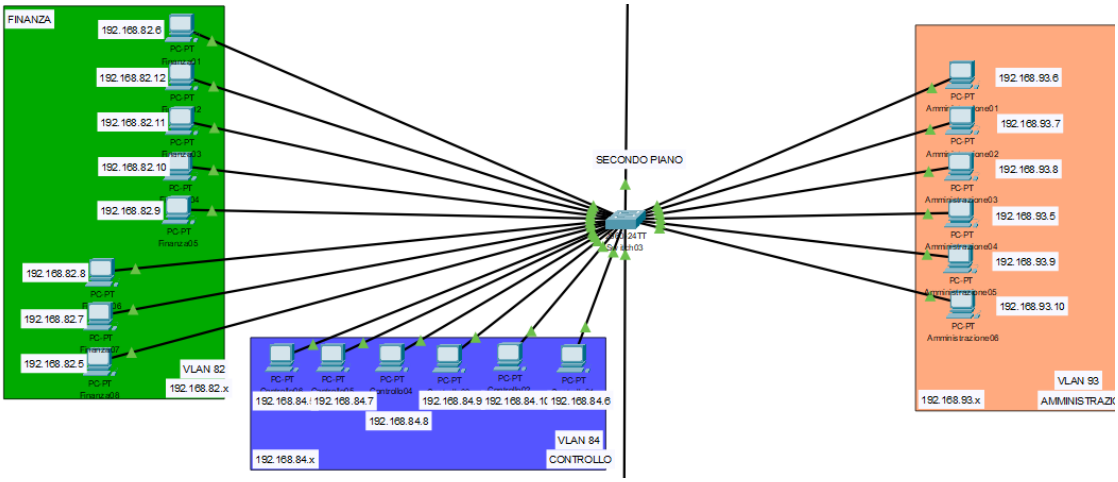


Secondo piano

Il cuore finanziario e burocratico della Compagnia Theta.

Reti e Componenti: 20 PC su Switch dedicato. Viene implementato il primo **IDS/IPS (1)** per la sorveglianza.

- **VLAN82/Finanza:** 192.168.82.x/24
- **VLAN84/Controllo:** 192.168.84.x/24
- **VLAN93/Amministrazione:** 192.168.93.x/24

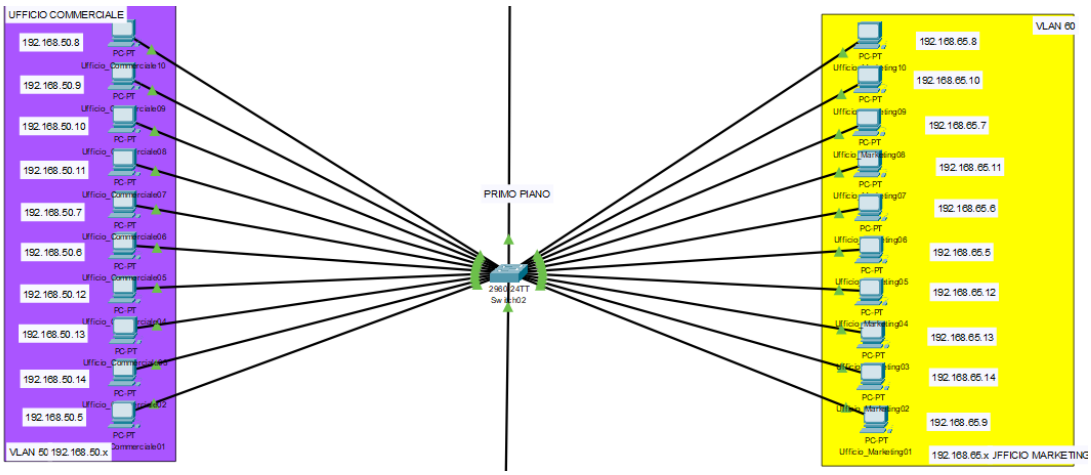


Primo piano

Questo piano è dedicato all'acquisizione di clientela e alla promozione dei prodotti.

Lo Switch serve i 20 PC con una VLAN unificata per promuovere la collaborazione tra i due team.

- VLAN50/Ufficio Commerciale:192.168.50.x/24
- VLAN60/Ufficio Marketing: 192.168.60.x/24

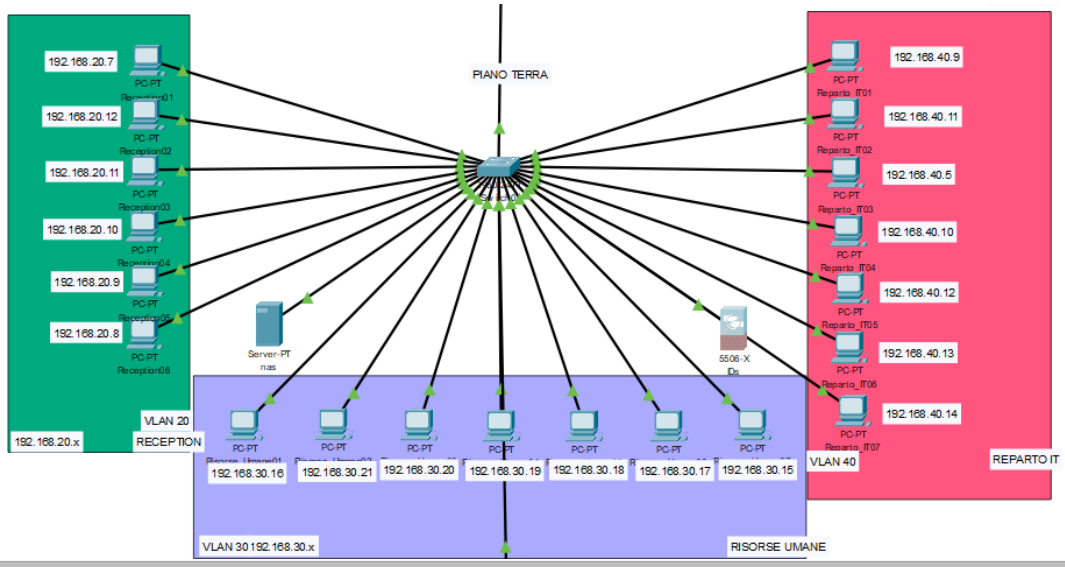


Piano Terra: IT, Risorse Umane e CED

Il Piano Terra è il fulcro operativo e logistico dell'azienda.

Reti e Componenti: Lo Switch di piano collega 20 PC. Il **Router** e il **NAS** sono qui posizionati.

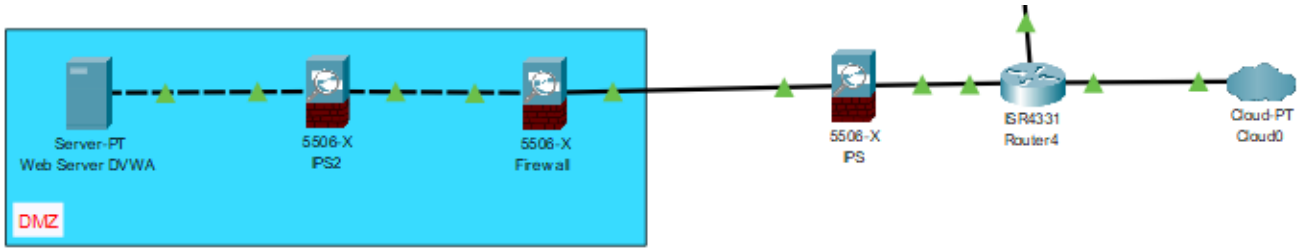
- **VLAN 20/Reception:** 192.168.20.x/24.
- **VLAN 30/Risorse Umane:** 192.168.2.x/24.
- **VLAN 40/Reparto IT:**192.168.40.x/24.



DMZ

La strategia di sicurezza di **VT Firewall S.r.l.** adotta una configurazione di **Difesa in Profondità (Defense in Depth)** per il perimetro esterno, garantendo che i servizi accessibili da Internet siano completamente isolati dalla rete interna critica.

Struttura della DMZ



La Zona Demilitarizzata (DMZ), identificata nell'immagine dall'area azzurra, è l'ambiente isolato dove risiede il servizio Web aziendale. Questa configurazione è progettata per contenere un'eventuale compromissione del server, impedendone la propagazione alla LAN interna.

Componente	Ruolo e Funzione di Sicurezza
Web Server DVWA	Server designato per l'hosting delle applicazioni pubbliche della Compagnia Theta
Firewall (Esterno)	Primo livello di difesa della DMZ. Controlla il traffico in entrata e in uscita, applicando politiche rigorose che consentono l'accesso alla DMZ solo sulle porte strettamente necessarie.
IPS (Interno DMZ)	Livello di sicurezza aggiuntivo. Il suo ruolo è monitorare il traffico tra il server e il Firewall Esterno, bloccando attivamente exploit noti o tentativi di scansione/attacco diretti al server, prima ancora che il traffico raggiunga la rete interna.

Connessione alla Rete Interna e al Router Core

Il traffico proveniente dalla DMZ è sottoposto a un ulteriore livello di ispezione prima di raggiungere la rete interna gestita dal **Router Core (ISR 4331)**:

- **Punto di Filtraggio:** L'ultima Appliance di sicurezza prima del Router Core è un **Intrusion Prevention System (IPS)** aggiuntivo.
- **Router Core (ISR 4331):** Rappresenta il punto di aggregazione finale. L'interfaccia verso la DMZ è configurata con regole di **ACL e Routing** che limitano severamente qualsiasi traffico in ingresso dalla DMZ verso le VLAN sensibili, completando così la strategia di isolamento a triplo strato.

Questa architettura garantisce che il Web Server, pur essendo un punto di esposizione necessario, non possa essere utilizzato come trampolino di lancio per attaccare la LAN interna.

3.3 Posizionamento e Ruolo dei Componenti Critici

Ogni dispositivo chiave è stato posizionato strategicamente per garantire che l'architettura rispetti il principio di sicurezza perimetrale e interna, in ottemperanza ai requisiti di progetto.

- **Firewall Perimetrale Esterno e DMZ:** Il **Firewall (5506-X)** è il *gatekeeper* primario, posizionato tra la connessione Internet (Cloud PT) e l'intera infrastruttura.
    - **Zona Demilitarizzata (DMZ):** Il **Web Server DVWA** risiede in questa zona isolata per ospitare servizi pubblici e contenere un eventuale attacco, impedendone la propagazione diretta alla LAN interna. Come si evince dal diagramma, la DMZ è ulteriormente protetta da dispositivi di sicurezza interni .
- **Router Centrale :** Posizionato nel **CED al Piano Terra** , funge da *Core Distribution*, aggregando i sei Switch di piano.
    - Il Router è il punto di controllo cruciale per il traffico interno, in quanto applica le **Access Control Lists (ACL)** necessarie a segregare logicamente le VLAN (e quindi i reparti).
- **NAS (Network Attached Storage):** Posizionato al Piano Terra, è collegato allo Switch di piano in prossimità del Router Core.
    - Questo posizionamento rende l'accesso al NAS facilmente controllabile dalle ACL impostate sul Router , garantendo che solo i reparti autorizzati (come Finanza o R&S) possano accedere ai dati sensibili archiviati.
- **Switch di Piano (n. 6):** Uno Switch è dedicato a ciascuno dei sei piani , garantendo la connettività per le 20 postazioni utente per piano.
    - Ogni Switch abilita la **segregazione logica in VLAN** (Topologia Logica) per separare i reparti in base alla sensibilità dei dati (es. VLAN Amministrazione, VLAN R&S).
- **Sistema IDS (Intrusion Detection System) Interno:** In linea con i requisiti di progetto (tre IDS/IPS totali), un sensore **IDS** è collegato strategicamente allo **Switch del Piano Terra**.
    - Questo IDS è deputato al monitoraggio dei flussi di traffico che transitano attraverso il *Core* della rete (Piano Terra), fornendo sorveglianza e allarme in tempo reale su tentativi di accesso non autorizzati o anomalie interne al perimetro LAN.

3.4 Strategia di Segmentazione e Indirizzamento (Subnetting)

La strategia di segmentazione logica implementata, basata sull'analisi della sensibilità dei dati, richiede una configurazione precisa a livello del Router Core L3 per gestire il traffico inter-VLAN e l'assegnazione degli indirizzi.

Per la gestione dinamica degli indirizzi IP sui 120 host (PC) aziendali, si è adottato il protocollo **DHCP (Dynamic Host Configuration Protocol)**, configurato direttamente sul Router L3 (o su un Server DHCP dedicato) per servire le subnet di ciascuna VLAN.

Punti Tecnici Chiave della Configurazione:

1. **Disattivazione dell'Interfaccia Fisica:** L'interfaccia fisica del Router collegata allo Switch (es. GigabitEthernet0/0) viene disattivata (no ip address) per garantire che l'instradamento avvenga esclusivamente sulle sottoreti logiche (Sub-interfacce).
2. **Creazione delle Sottoreti Logiche:** Per ogni VLAN creata (VLAN n), si configura una *sotto-interfaccia* logica (GigabitEthernet0/0.n) che agisce da Default Gateway per quella specifica rete (es. 192.168.30.1 per la VLAN 30).
    - **Identificazione VLAN:** Il comando encapsulation dot1Qn tagga il traffico sulla porta *Trunk* con l'ID della VLAN corrispondente.
3. **Configurazione DHCP Relay:** Sui dispositivi che non ospitano direttamente il server DHCP (come le Sub-interfacce del Router L3, se il server è esterno), viene configurato il comando **ip helper-address**. Questo indirizza le richieste DHCP degli host verso l'indirizzo IP del server DHCP centrale, garantendo che tutti i PC ricevano il corretto indirizzo IP, la Subnet Mask e il Default Gateway per la loro specifica rete.

3.5 Accorgimenti e Best Practice per la VLAN

Per garantire la coerenza della configurazione sugli Switch di accesso (quelli di piano), si è seguita la seguente prassi:

- **Uniformità delle VLAN:** Ogni Switch è configurato per ospitare la VLAN di competenza di quel piano. Ad esempio, lo *Switch Piano Terra* gestirà la VLAN IT/HR, mentre lo *Switch Primo Piano* gestirà la VLAN Commerciale/Marketing, in linea con la segmentazione fisica/logica.
- **Gestione dei Trunk:** Le porte dello Switch collegate al Router L3 (Trunk) devono essere configurate correttamente per trasportare il traffico di tutte le VLAN necessarie, incluse quelle utilizzate per la gestione della rete.

4 Ambiente di Virtualizzazione e Implementazione del Lab

Per garantire un'efficace e sicura fase di testing del progetto di rete per la Compagnia Theta, VT Firewall S.r.l. ha implementato un ambiente di laboratorio virtuale. L'utilizzo della virtualizzazione ha permesso di isolare completamente le attività di testing dalla rete di produzione e di replicare con precisione l'infrastruttura di sicurezza.

4.1 Piattaforma e Macchine Virtuali (VM)

L'ambiente di lab è stato costruito sulla piattaforma **VirtualBox**. Le seguenti macchine virtuali sono state ospitate per simulare i ruoli critici nell'architettura finale:

- **Web Server Target (Metasploitable 2):**
  - **Ruolo:** Simula il Web Server aziendale esposto, con a bordo l'applicazione vulnerabile **DVWA (Damn Vulnerable Web Application)**, come richiesto dalle specifiche di progetto.
  - **Posizionamento Logico:** Questa VM è stata posizionata logicamente nella **Zona Demilitarizzata (DMZ)**, isolata dal resto della rete interna per testare le regole del Firewall perimetrale.
- **Macchina di Attacco (Kali Linux/OS di Testing):**
  - **Ruolo:** Ha ospitato i **tool di scansione Python proprietari** sviluppati per il testing (Port Scanner e Analizzatore Verbi HTTP).
  - **Posizionamento Logico:** La macchina di testing è stata configurata in una subnet separata per simulare sia l'attacco dall'esterno (Internet, per la DMZ) sia il testing interno (dalla LAN aziendale) verso i dispositivi critici.

4.2 Isolamento e Connettività

La configurazione delle schede di rete virtuali (Virtual Network Interfaces) ha assicurato l'isolamento completo:

1. **Rete di Testing:** Tutti i dispositivi critici (Router virtuale, Firewall virtuale, Web Server, Macchina Kali) sono stati interconnessi in una **rete virtuale isolata** per garantire che il testing non interferisse con altre reti.
2. **Web Server in DMZ:** La VM del Web Server è stata connessa al Router/Firewall in modo da replicare esattamente l'indirizzo IP designato per la DMZ, convalidando l'efficacia delle **Access Control Lists (ACL)** e delle regole di NAT.

Questo approccio metodologico ha garantito che i risultati dei test riportati nella Sezione 6 siano il più possibile rappresentativi del comportamento atteso nell'ambiente di produzione della Compagnia Theta.

4.3 Implementazione dei Servizi di Rete Critici (Integrazione pfSense)

Per la gestione avanzata del Firewall perimetrale e dei servizi di rete essenziali, l'architettura si avvale della piattaforma **pfSense**.

Ruolo di pfSense nell'Architettura

- **Firewall Perimetrale Avanzato:** pfSense funge da Firewall perimetrale tra la Rete Interna (LAN) e la connessione Internet, gestendo la **DMZ** e applicando le regole di filtraggio.
- **Routing e NAT:** È configurato per gestire il **Network Address Translation (NAT)**, mascherando gli indirizzi IP privati della LAN dietro l'unico indirizzo IP pubblico, garantendo così che le comunicazioni esterne siano sicure.
- **Servizi di Rete:** Oltre al filtraggio, pfSense è la piattaforma scelta per l'implementazione dei servizi di rete fondamentali.

4.4 Configurazione e Isolamento Logico degli Adattatori di Rete

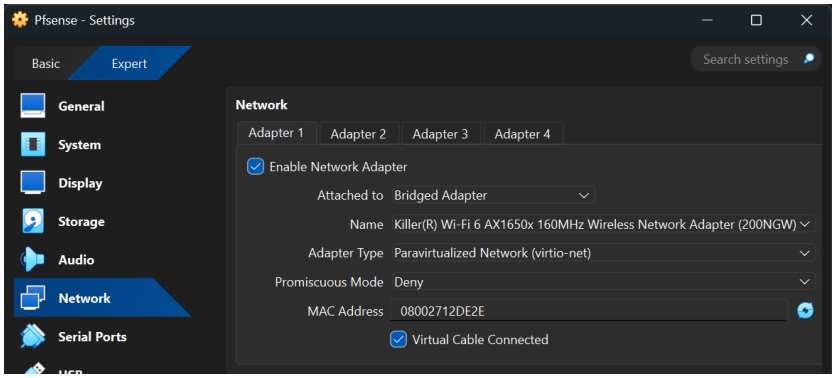
L'ambiente di lab è stato segmentato utilizzando la funzione **Internal Network** della piattaforma di virtualizzazione per creare domini logici isolati e strettamente controllati:

- **Rete DMZ (Zona Demilitarizzata):** È stata creata una rete virtuale denominata DMZ. Questa rete è usata esclusivamente per connettere l'interfaccia DMZ del Firewall (pfSense) e il Web Server target.
- **Rete Interna (LAN):** È stata creata una rete virtuale denominata Interna. Questa rete connette l'interfaccia LAN del Firewall (pfSense) alla Macchina di Testing (Kali Linux), simulando la LAN aziendale.

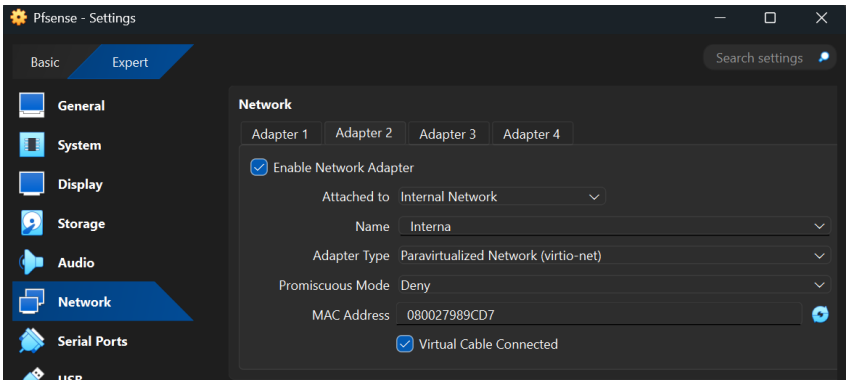
Dettaglio della Configurazione degli Adattatori

La configurazione delle interfacce per i tre dispositivi virtuali principali è stata essenziale per stabilire i confini di sicurezza logica:

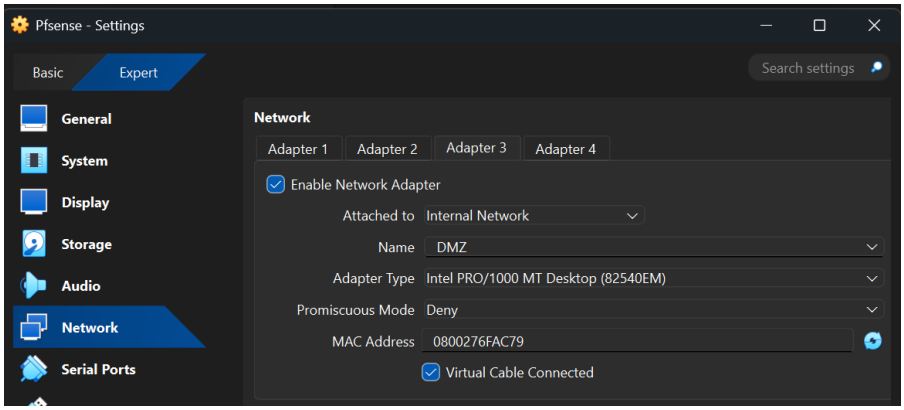
1. **Firewall Perimetrale (pfSense):**
    - **WAN Connection:** Un adattatore è stato configurato in modalità **Bridged Adapter** per simulare la connessione alla rete esterna (Internet/WAN).



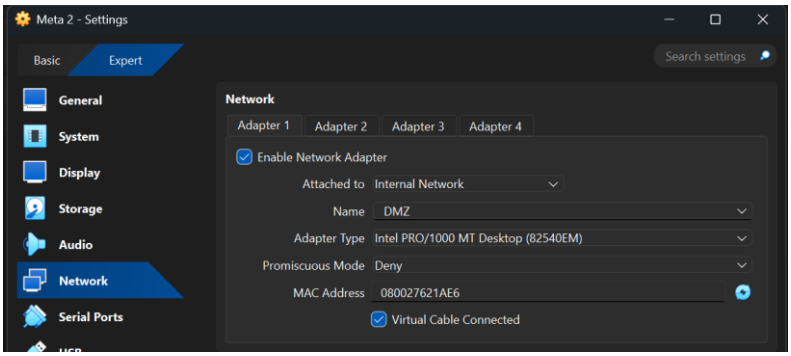
    - **LAN Connection:** Un secondo adattatore è stato collegato alla rete virtuale denominata **Interna**, stabilendo l'interfaccia di gestione del traffico per la LAN aziendale



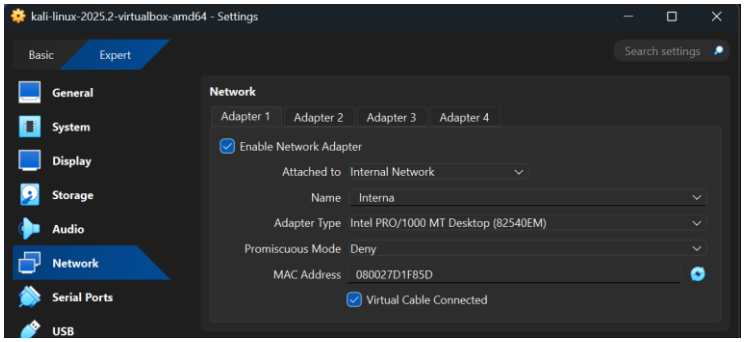
- **DMZ Connection:** Un terzo adattatore è stato collegato alla rete virtuale **DMZ**, dedicandolo all'isolamento del server pubblico.



2. **Web Server Target (Metasploitable 2):**
   - Il Web Server è stato configurato con un unico adattatore di rete collegato alla rete virtuale denominata **DMZ**. Questo lo isola da qualsiasi altra rete e ne assicura la raggiungibilità solo tramite il Firewall.



3. **Macchina di Testing (Kali Linux):**
   - La macchina utilizzata per eseguire i tool Python custom è stata collegata alla rete virtuale denominata **Interna**. Questo posizionamento la rende logicamente parte della LAN simulata, permettendo di verificare le politiche di sicurezza interna.

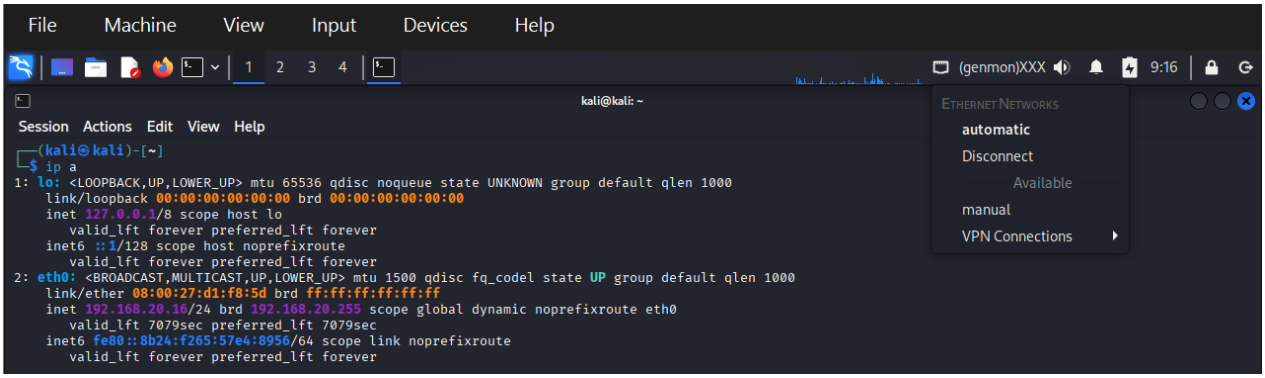


L'utilizzo della modalità **Internal Network** per le interfacce LAN e DMZ garantisce che il traffico rimanga confinato all'ambiente virtuale, isolando l'architettura simulata da qualsiasi rete fisica esterna (ad eccezione dell'adattatore WAN di pfSense). Questa configurazione robusta stabilisce il campo di prova per i successivi test di sicurezza.

Configurazione Scheda di Rete Kali Linux

La Macchina di Testing è stata configurata con un'interfaccia di rete connessa alla rete virtuale **Interna** e ha ricevuto un indirizzo IP dinamico coerente con la subnet LAN gestita dal Firewall pfSense.

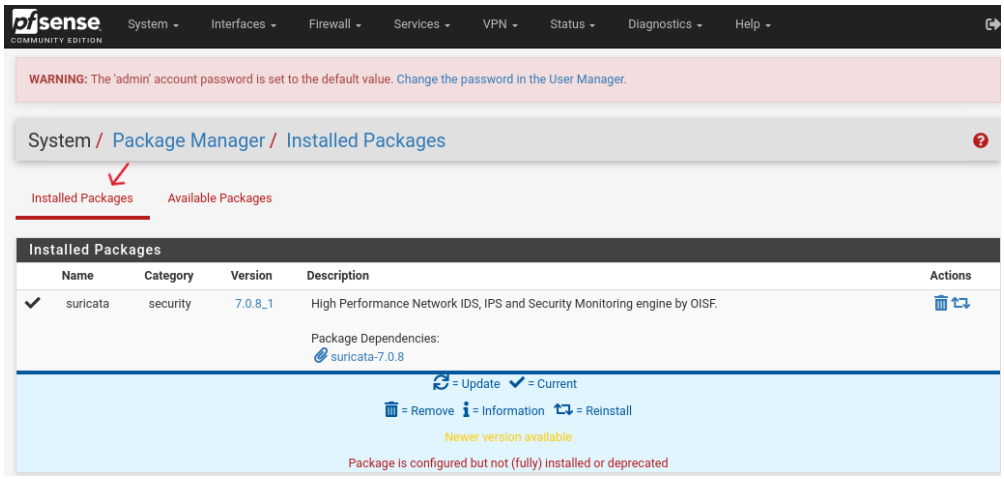
- **Verifica Assegnazione IP:** L'esecuzione del comando ip a sulla macchina Kali Linux conferma che l'interfaccia **eth0** è operativa e ha ricevuto l'indirizzo IP **192.168.20.16/24**. Questo indirizzo appartiene alla subnet (192.168.20.0/24) della LAN interna, convalidando il corretto posizionamento logico della macchina per eseguire i test.



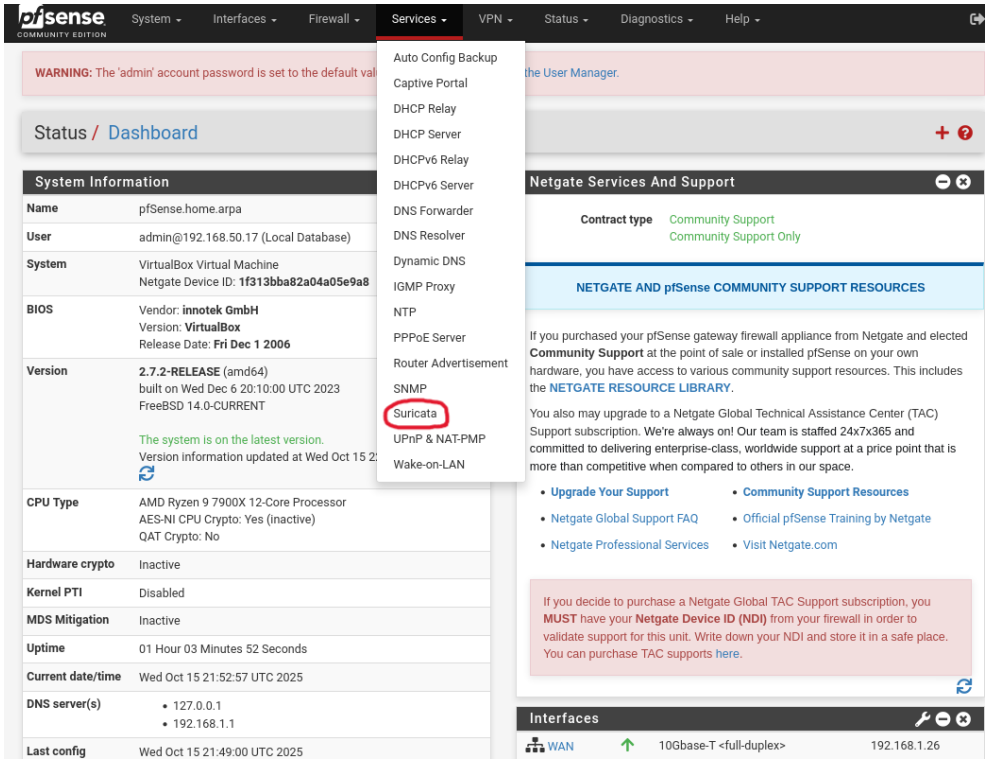
5 Integrazione e Monitoraggio della Sicurezza di Rete (Suricata)

5.1 Dettagli Implementativi del Sistema di Rilevamento delle Intrusioni (IDS)

Il sistema di Network Intrusion Detection System (NIDS) è stato implementato e configurato utilizzando il pacchetto Suricata, motore ad alte prestazioni per la sicurezza di rete, integrato nella piattaforma pfSense. La versione installata di Suricata è la 7.0.8_1.



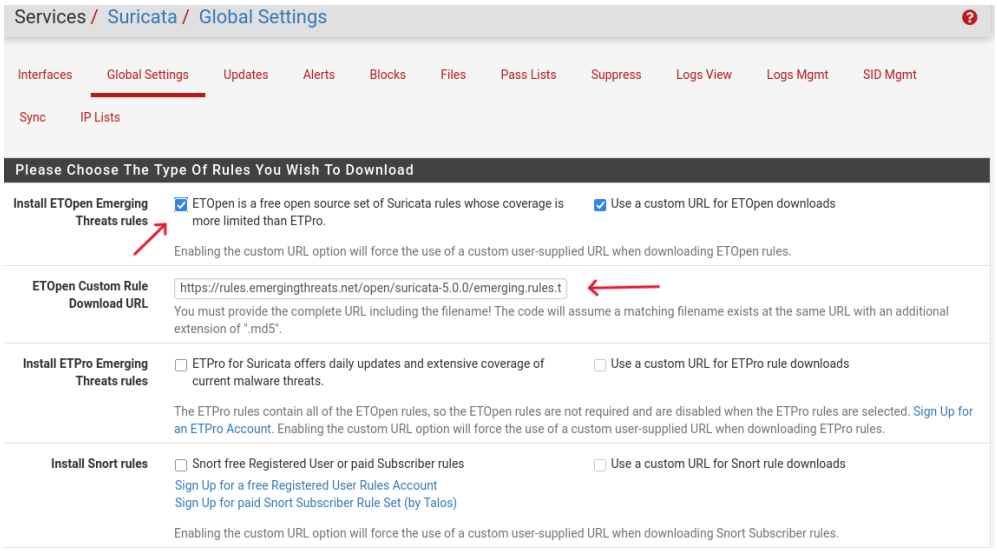
L'accesso alla configurazione e al monitoraggio di Suricata avviene tramite il menu dei servizi di pfSense, garantendo un controllo centralizzato sull'attività di ispezione del traffico.



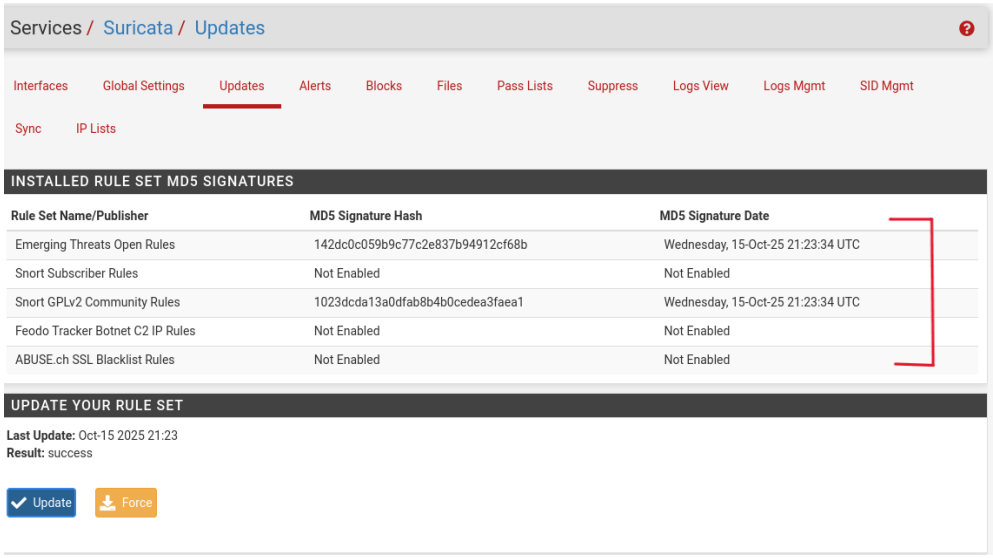
5.2 Configurazione e Aggiornamento dei Set di Regole

La strategia di difesa si basa sull'utilizzo di set di regole aggiornati per garantire la massima copertura contro le minacce note. È stato scelto di installare le **Emerging Threats Open Rules (ETOpen)**, una risorsa *open source* essenziale per un'efficace sorveglianza del traffico.

La configurazione prevede l'utilizzo di una URL custom per il download delle regole, assicurando la corretta localizzazione della risorsa di aggiornamento: https://rules.emergingthreats.net/open/suricata-5.0.0/emerging.rules.tar.gz.



Gli aggiornamenti del set di regole sono eseguiti regolarmente. L'ultima operazione di *update* è stata completata con successo in data 15 ottobre 2025, come confermato dal *timestamp* delle firme MD5.



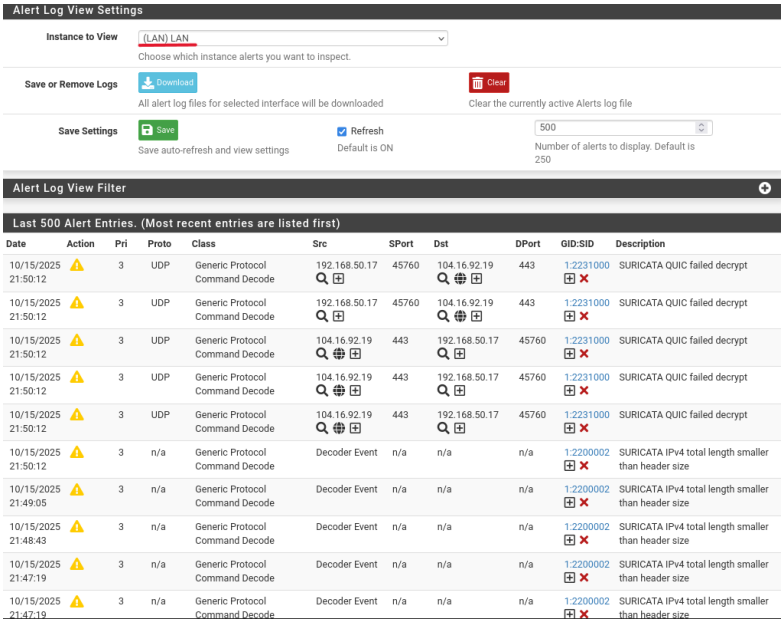
5.3 Stato Operativo dell'Interfaccia LAN

Suricata è stato abilitato e reso operativo sull'interfaccia **LAN (vtnet1)**. Lo stato corrente del servizio è confermato come **Attivo** (simbolo di spunta verde), indicando che il monitoraggio del traffico sulla rete locale è in corso. La modalità di blocco (*Blocking Mode*) è attualmente **Disabilitata**, configurando Suricata come puro IDS (Intrusion Detection System) per la sola registrazione degli avvisi, senza intervento attivo sul flusso di dati.

5.4 Analisi dei Log di Allerta Recenti

La fase di monitoraggio iniziale ha generato una serie di avvisi registrati nel *Log View* di Suricata. L'analisi dei *500 Alert Entries* più recenti, elencati in ordine cronologico inverso, rivela la presenza di due categorie principali di avvisi:

1. **Avvisi GID:1:SID:2231000** ("SURICATA QUIC failed decrypt"):
    - Questi avvisi sono stati registrati a partire dalle 21:50:12 del 15/10/2025 e riguardano tentativi falliti di decifratura di pacchetti del protocollo QUIC. Tali eventi sono tipici in contesti in cui il traffico QUIC (spesso utilizzato da Google Chrome e da servizi Google) è ispezionato senza la disponibilità della chiave di sessione, e non indicano necessariamente una minaccia attiva, ma piuttosto una limitazione nell'ispezione profonda del traffico crittografato.
2. **Avvisi GID:1:SID:2200002** ("SURICATA IPv4 total length smaller than header size"):
    - Questi allarmi sono stati registrati a partire dalle 21:47:19 del 15/10/2025 e indicano un *Decoder Event*. Questo specifico avviso segnala pacchetti IP con una dimensione totale dichiarata inferiore alla dimensione minima dell'intestazione IPv4. Questo può essere il risultato di un *fuzzing* di rete, di un tentativo di evasione IDS o semplicemente di pacchetti malformati, e merita ulteriori indagini.



6 Metodologia di Testing e Sviluppo di Tool Custom

6.1 Obiettivi e Metodologia di Verifica

Prima della messa in produzione, **VT Firewall S.r.l.** ha condotto una fase di *Testing della Rete* mirata a convalidare le politiche di sicurezza e identificare potenziali vulnerabilità sull'infrastruttura implementata in ambiente simulato. L'obiettivo primario è stato il *Web Server (DVWA di Metasploitable)*, in quanto esposto nella Zona Demilitarizzata (DMZ) e quindi a rischio di attacchi esterni.

La metodologia ha richiesto lo sviluppo di **tool di analisi proprietari in linguaggio Python**, in ottemperanza ai requisiti di progetto.

6.2 Tool 1: Scanner Custom per la Scansione delle Porte

Per verificare l'accessibilità dei servizi sui dispositivi di rete (come il Web Server e i dispositivi di gestione interna), è stato sviluppato uno scanner di porte personalizzato.

- **Funzionalità:** Il tool, scritto in Python, accetta in input un indirizzo IP e un *range* di porte da scansionare (es. 1-1024).
- **Obiettivo di Sicurezza:** Verificare se porte di servizio note e potenzialmente insicure (es. Telnet/23, FTP/21, SSH/22) siano esposte in modo inopportuno, specialmente sul Web Server Metasploitable.
- **Output:** Genera una lista delle porte con il relativo stato di accessibilità (Aperta/Chiusa).

Analisi del codice realizzato

Blocco 1: **Importazione e Mappatura dei Servizi**

Questo blocco iniziale stabilisce i moduli necessari per le operazioni di rete e crea la mappa di riferimento per l'identificazione dei servizi standard (es. HTTP, FTP).

```
1   import socket
2   import sys
3
4   #Variabili per Ip host e porte da scannerizzere
5   # target_Host = "192.168.50.16"
6   # ports_to_Scan = [80,25,53,44444]
7   # TIMEOUT = 1
8
9   #Mappatura porte comuni
10  ports_Map = {21: "FTP", 22: "SSH", 23: "Telnet", 25: "SMTP",
11      53: "DNS", 80: "HTTP", 443: "HTTPS"}
```

Descrizione Tecnica: Si importano le librerie essenziali: **socket** per gestire le connessioni TCP/IP e **sys** per controllare il flusso del programma (es. uscita forzata in caso di errore). Il dizionario ports_Map è una *best practice* per convertire i numeri di porta grezzi in nomi di servizio leggibili nel report finale.

Blocco 2: Funzione Principale di Scansione (scansiona_porte)

Questo blocco contiene la logica core del tool. La funzione tenta di stabilire una connessione TCP con l'host e la porta specificati.

```python
def scansiona_porte (host , port) :
    print(f"\n--- Inizio scansione su IP:{host}---\n")
    report_data_port_scanner=[]
    if not host :
        print("ERRORE: Host non specificato")
        return []

    for porta in ports_to_Scan :
        risultato = {"Host": host , "Porta" : porta ,"Servizio" : ports_Map.get(porta,"Unknown"),"Stato" : "Errore","Messaggio" :"printRisultato" }
        s= None
        try :
            s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
            s.settimeout(5)
            stato = s.connect_ex((host,porta))
            s.close() #Chiusura socket
            if(stato == 0):
                risultato["Stato"] = "Aperta"
                risultato["Messaggio"] = "Accesso consentito."

            else:
                risultato["Stato"] = "Chiusa"
                risultato["Messaggio"] = "Porta chiusa o esiste una regola per tale porta all'interno del Firewall"


        except (socket.gaierror,socket.error) as error :
            risultato["Stato"] = "Errore nell'host o nella connessione"
            risultato["Messaggio"] = f"Impossibile connettersi all'host {error}"
        finally :
            if s:
                s.close()

        print(f"Porta {risultato['Porta']:<5} ({risultato['Servizio']:<10}) : {risultato['Stato']:<15}")
        #Usando : inizia una specifica del formato < specifica allineamento e il numero quanto spazio deve occupare in caratteri
        report_data_port_scanner.append(risultato)


    return report_data_port_scanner
```

Descrizione Tecnica: La funzione itera sulla lista delle porte da testare. Per ciascuna porta:

1. Viene creato un socket TCP/IPv4 (AF_INET, SOCK_STREAM).
2. Viene impostato un timeout di 5 secondi (s.settimeout(5)) per gestire le porte filtrate senza bloccare l'esecuzione.
3. Si utilizza s.connect_ex(), che tenta la connessione in modo *non bloccante*.
4. Se il valore di ritorno è 0, la porta è classificata come Aperta (servizio in ascolto). Altrimenti, è Chiusa (o filtrata dal Firewall).
5. Il blocco try...except garantisce la gestione degli errori, in particolare quelli relativi all'host o alla rete. I risultati sono aggregati in una lista di dizionari per la successiva fase di *reporting*.

Blocco 3: Blocco di Esecuzione e Output (Interfaccia Utente)

Questo blocco gestisce l'interazione con l'analista (input dei parametri) e formatta l'output finale dei risultati per il report.

```python
if __name__ == "__main__" :

    input_host = input("Inserisci l'IP del server target (es. 192.168.1.1): ")
    input_ports_str = input("Inserisci le porte da scansionare, separate da virgola (es. 80,22,443): ")
    ports_to_Scan = []
    try :
        ports_to_Scan=[int(p.strip()) for p in input_ports_str.split(',') if p.strip().isdigit()]
    except Exception as error :
        print(f"ERRORE: Impossibile convertire le porte in numeri interi {error}")
        sys.exit(1)
    if not ports_to_Scan :
        print("Nessuna porta valida inserita . Uscita")
        sys.exit(1)
    report_scanner = scansiona_porte(input_host,ports_to_Scan)

    print("\n Dati per Report ")
    for item in report_scanner :
        if "Aperta" in item["Stato"]:
            print(f"Porta Aperta: {item['Porta']} ({item["Servizio"]}) ---> Sicurezza : {item["Messaggio"]}")
        elif "Chiusa" in item["Stato"] :
            print(f"Porta Chiusa : {item["Porta"]} ({item["Servizio"]}) ")
        elif "Errore" in item["Stato"]:
            print(f"Porta {item['Porta']} ({item['Servizio']}) ---> {item['Stato']}: {item['Messaggio']}")
```

Descrizione Tecnica: Il blocco di controllo if __name__ == "__main__" gestisce l'interfaccia a linea di comando.

1. **Input Validation:** Viene richiesto l'IP e una lista di porte, che sono poi validate e convertite in interi. In caso di input non validi, il programma si chiude in modo controllato (sys.exit(1)).
2. **Generazione Output:** La sezione finale genera un output strutturato che riassume lo stato di ogni porta. Questo output, formattato per distinguere chiaramente le porte **Aperte** da quelle **Chiuse**, è ciò che viene utilizzato direttamente come prova documentale per il report di sicurezza (Sezione 5).

6.3 Tool 2: Analizzatore dei Verbi HTTP

Per valutare la configurazione di sicurezza del Web Server, è stato necessario analizzare i metodi HTTP abilitati, che potrebbero consentire azioni dannose come il caricamento non autorizzato di file o la sovrascrittura di contenuti.

- **Funzionalità:** Il tool Python è stato progettato per inviare richieste specifiche a un determinato *path* del Web Server (in questo caso, l'applicazione phpMyAdmin su Metasploitable) e verificarne le risposte.
- **Obiettivo di Sicurezza:** Individuare l'abilitazione di verbi HTTP potenzialmente pericolosi come PUT, DELETE o OPTIONS, la cui presenza non necessaria rappresenta una vulnerabilità di configurazione.
- **Output:** Restituisce una lista dei metodi HTTP abilitati sul path specificato.

Analisi del codice realizzato

```
1   import http.client
2
3   # host = "192.168.20.10"
4   # port = 80 #Porta di default per HTTP
5   # path= "/dvwa/"
6
7   # payload_POST = b"username=admin&password=password&Login=Login"
8   # payload_PUT = b"data=test_put_update"
9   content_type = "application/x-www-form-urlencoded" #Header per POST e PUT
```

Descrizione Tecnica:

Viene importato il modulo http.client per gestire le richieste HTTP a livello di protocollo. La variabile content_type definisce l'header standard utilizzato per inviare i dati di un form, necessario per testare i metodi POST e PUT con un payload (come i dati di login).

Blocco 2: Funzione Principale di Verifica (verifica_Http)

Questa funzione contiene la logica centrale, gestendo l'elaborazione del payload, l'iterazione sui verbi HTTP e la logica di gestione della connessione e della risposta.

```
def verifica_Http(host,port,path,custom_payload) :
    report_data_http = []
    verbi_test = ["GET","DELETE","POST","PUT"]
    payload_bytes= None
    payload_sent_str = "Non presente"

    if custom_payload :
        try :
            payload_bytes = custom_payload.encode('utf-8')
            payload_sent_str = custom_payload
        except Exception :
            payload_sent_str = "Errore Codifica Payload"

    for verbo in verbi_test :
        #Variabili per payload e headers
        body = None
        headers={}
        payload_da_usare = "Non presente"
        if verbo == "POST" or verbo == "PUT" :
            if payload_bytes:
                body = payload_bytes
                headers = {"Content-Type" : content_type}
                payload_da_usare = payload_sent_str
            else :
                payload_da_usare="POST/PUT richiesto,payload vuoto"

        risultato = {"Verbo":verbo , "URL_Test":f"http://{host}:{port}{path}","Codice_Stato":"non presente","Dettaglio_Risposta" : "non presente" , "Payload" : payload_da_usare}
```

Descrizione Tecnica: La funzione inizializza la lista dei verbi da testare (GET, DELETE, POST, PUT). Gestisce l'encoding del payload fornito dall'utente in *bytes* (necessario per l'invio HTTP). Il loop principale itera su ogni verbo, preparando dinamicamente gli header e il body per i metodi POST e PUT prima di inviare la richiesta.

Blocco 3: Gestione della Connessione, Richiesta e Risposta

Questo blocco è responsabile della comunicazione effettiva con il Web Server e dell'analisi della risposta ricevuta (Status Code e Reason).

```
        conn =None
        try :
            conn = http.client.HTTPConnection(host,port,timeout=5)
            conn.request(verbo,path,body=body,headers=headers)
            response = conn.getresponse()
            risultato["Codice_Stato"] = response.status
            if 200 <= risultato["Codice_Stato"] < 400 :
                risultato["Dettaglio_Risposta"] = f"Successo : {response.reason}"
            elif verbo == "OPTIONS" and response.status == 200:
                risultato["Dettaglio_Risposta"] = f"Metodi Abilitati : {response.getheader('Allow','Non trovato')}"
            else :
                risultato["Dettaglio_Risposta"] = response.reason

            print(f"{verbo:<8} : Stato{risultato['Codice_Stato']} - {risultato['Dettaglio_Risposta'][:40]}...")
        except ConnectionRefusedError:
            risultato["Dettaglio_Risposta"] = "Connessione Fallita"
            print(f"{verbo}:{risultato['Dettaglio_Risposta']}")
        except Exception as error :
            risultato ["Dettaglio_Risposta"] = f"Errore generico {str(error)}"
        finally :
            if conn :
                 conn.close()


        report_data_http.append(risultato)
```

Descrizione Tecnica: Viene stabilita la HTTPConnection con un timeout di 5 secondi. Il metodo conn.request() invia la richiesta HTTP con il verbo, il path, e il body/headers opportuni. L'output viene classificato: le risposte tra 200 e 399 sono segnalate come Successo (che include anche i reindirizzamenti 302), mentre altre risposte (es. 404 Not Found, 405 Method Not Allowed) vengono registrate tramite il campo response.reason. Il blocco finally assicura che la connessione venga sempre chiusa.

Blocco 4: Interfaccia Utente e Output Finale

Questo blocco di esecuzione (if __name__ == "__main__":) gestisce l'input interattivo e formatta l'output dei risultati in modo dettagliato per il report.

```
if __name__ == "__main__":
    input_host = input(f"Inserisci l'IP del server(default : 192.168.20.10): ")
    input_port = input("Inserisci la porta(default : 80)")
    try :
        input_port=int(input_port)
    except ValueError :
        input_port=80
    input_path = input("Inserisci il Path(es. /dvwa/ -default : /):")
    custom_payload = input("Inserisci il Payload(Es. username=a&password=b)o lascia vuoto: ")
    risultati_test = verifica_Http(input_host,input_port,input_path,custom_payload)

    for risultato in risultati_test:
        payload_disp = ""
        if risultato.get('Payload') not in ['Non presente',None,'POST/PUT richiesto, payload vuoto'] :
            payload_disp = f" (Payload: {risultato['Payload']})"
        print(f"[{risultato['Verbo']:<8}] Stato: {risultato['Codice_Stato']} | Dettaglio: {risultato['Dettaglio_Risposta']}{payload_disp}")
```

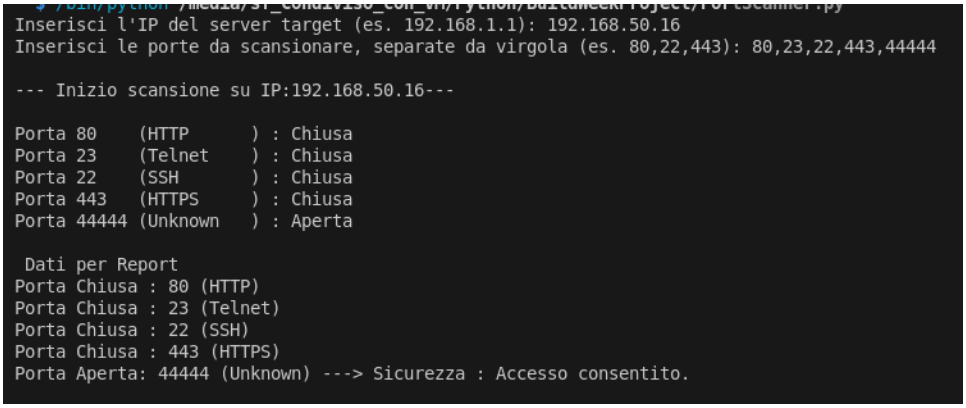
Descrizione Tecnica: L'interfaccia a riga di comando richiede all'analista i parametri chiave (IP, Porta, Path e Payload), essenziali per eseguire i test mirati sulla DMZ. La sezione finale stampa i risultati formattati, includendo lo **Status Code**, il **Dettaglio della Risposta** e, se applicabile, il **Payload** utilizzato, fornendo una prova documentale chiara delle risposte del Web Server.

7 Risultati dei Test e Analisi delle Vulnerabilità

La fase di testing è stata condotta per convalidare la configurazione di sicurezza del Web Server (DVWA di Metasploitable), posizionato nella Zona Demilitarizzata (DMZ).

7.1 Risultati della Scansione delle Porte

La scansione è stata eseguita utilizzando lo strumento proprietario Port Scanner sviluppato da VT Firewall S.r.l. Il test è stato indirizzato all'IP 192.168.50.16 (simulando un dispositivo di rete critico o il Web Server dopo la configurazione Firewall), testando porte standard e una porta alta casuale (44444).



Analisi e Rischio Rilevato

RISCHIO CRITICO RILEVATO: Servizio Sconosciuto sulla Porta 44444

I risultati dimostrano che i servizi standard ad alto rischio (SSH, Telnet) e i servizi web (HTTP, HTTPS) sono correttamente non accessibili, indicando che:

1. Le regole di filtraggio del Firewall stanno funzionando per le porte di default.
2. Il server non è esposto su servizi non necessari a basso numero di porta.

Tuttavia, il rilevamento della Porta 44444 come Aperta e con "Accesso consentito" è un segnale di allarme critico. La presenza di un servizio in ascolto su una porta alta e sconosciuta è tipica di:

- Backdoor: Un accesso remoto segreto lasciato da un attaccante.
- Servizio Caching/Amministrazione Non Documentato: Un'applicazione di gestione interna che non dovrebbe essere accessibile.

Raccomandazione Preliminare

È imperativo eseguire un'analisi approfondita (Service Fingerprinting) su questo servizio sconosciuto per identificarne lo scopo e bloccare immediatamente la porta 44444 su tutti i dispositivi di filtraggio (Firewall e/o ACL del Router) fino alla completa identificazione e bonifica della minaccia potenziale.

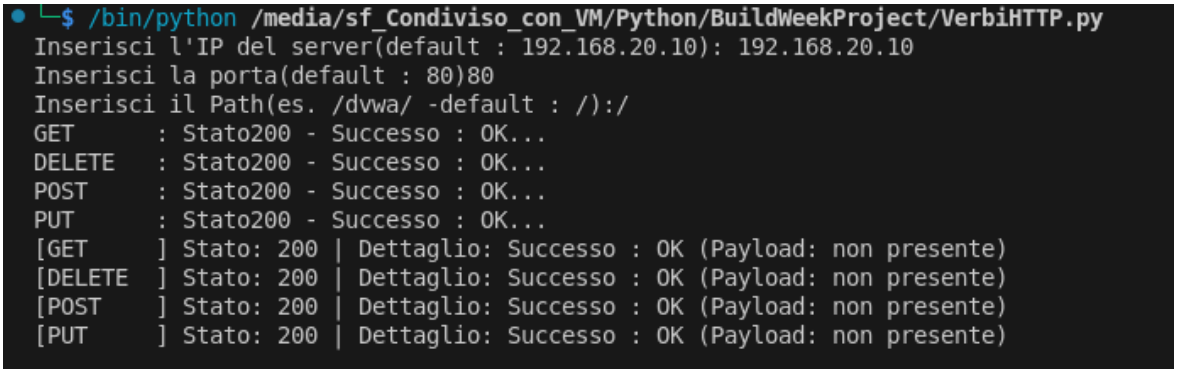
7.2 Risultati della Scansione dei VerbiHTTP

Il tool personalizzato in Python è stato eseguito contro l'indirizzo IP del **Web Server DVWA** (192.168.20.10), testando quattro diversi percorsi per identificare l'abilitazione di metodi HTTP potenzialmente rischiosi:

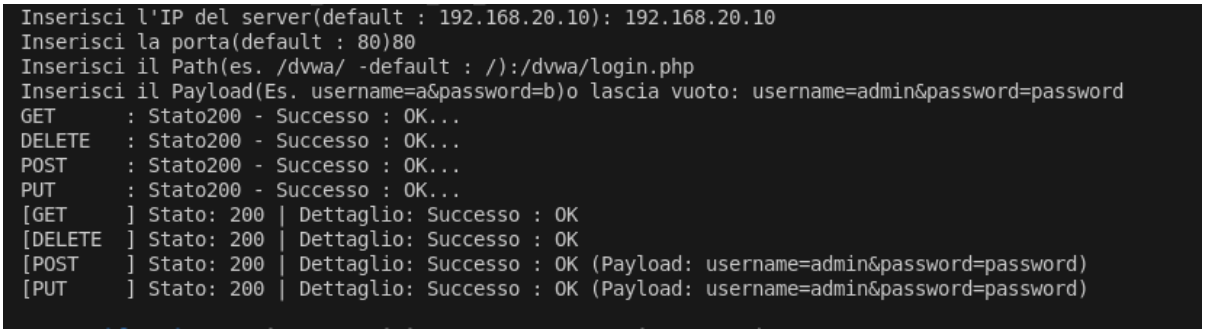
A. Risultati Dettagliati per Path

Di seguito, vengono presentati gli screenshot che documentano l'output dei test eseguiti sul Web Server, a riprova della metodologia e dei risultati ottenuti.

1. **Verifica sul Path di Root (/)** Questo test ha verificato i metodi abilitati sulla directory principale del server. L'output dimostra chiaramente che tutti i metodi chiave, inclusi i rischiosi PUT e DELETE, sono attivi e rispondono con successo (Stato 200 - OK).



2. **Verifica sulla Pagina di Login (/dvwa/login.php) con Payload** Questo test ha simulato una richiesta sulla pagina di autenticazione, mostrando che il server riconosce i metodi e risponde positivamente anche in presenza di dati di login (payload).



3. Verifica sul Path Base DVWA (/dvwa/) Questo test ha mostrato il comportamento del server sul path principale dell'applicazione, indicando un reindirizzamento (Status 302 - Found) per tutti i verbi, confermandone l'abilitazione.

```
└─$ /bin/python /media/sf_Condiviso_con_VM/Python/BuildWeekProject/VerbiHTTP.py
Inserisci l'IP del server(default : 192.168.20.10): 192.168.20.10
Inserisci la porta(default : 80)80
Inserisci il Path(es. /dvwa/ -default : /):/dvwa/
GET      : Stato302 - Successo : Found...
DELETE   : Stato302 - Successo : Found...
POST     : Stato302 - Successo : Found...
PUT      : Stato302 - Successo : Found...
[GET     ] Stato: 302 | Dettaglio: Successo : Found (Payload: non presente)
[DELETE  ] Stato: 302 | Dettaglio: Successo : Found (Payload: non presente)
[POST    ] Stato: 302 | Dettaglio: Successo : Found (Payload: non presente)
[PUT     ] Stato: 302 | Dettaglio: Successo : Found (Payload: non presente)
```

4. Verifica su Risorsa Inesistente (/risorsa) Questo test è stato eseguito per confrontare il comportamento del server su un path non esistente, confermando una gestione parzialmente corretta: i metodi **GET** e **POST** risultano in un 404 - Not Found, mentre i metodi potenzialmente pericolosi **DELETE** e **PUT** risultano in un 405 - Method Not Allowed.

```
└─$ /bin/python /media/sf_Condiviso_con_VM/Python/BuildWeekProject/VerbiHTTP.py
Inserisci l'IP del server(default : 192.168.20.10): 192.168.20.10
Inserisci la porta(default : 80)80
Inserisci il Path(es. /dvwa/ -default : /):/risorsa
GET      : Stato404 - Not Found...
DELETE   : Stato405 - Method Not Allowed...
POST     : Stato404 - Not Found...
PUT      : Stato405 - Method Not Allowed...
[GET     ] Stato: 404 | Dettaglio: Not Found (Payload: non presente)
[DELETE  ] Stato: 405 | Dettaglio: Method Not Allowed (Payload: non presente)
[POST    ] Stato: 404 | Dettaglio: Not Found (Payload: non presente)
[PUT     ] Stato: 405 | Dettaglio: Method Not Allowed (Payload: non presente)
```

RISCHIO CRITICO RILEVATO: Esposizione di Metodi Pericolosi

I test sul path di **Root (/)** e sulla pagina di **Login (/dvwa/login.php)** hanno confermato che il Web Server **abilita e risponde con successo (200 OK e 302 Found) ai metodi DELETE e PUT**.

1. **PUT Abilitato:** Questo metodo consente la creazione o la sovrascrittura di file sul server. Se un attaccante riuscisse a trovare una directory con permessi di scrittura, il server sarebbe vulnerabile al caricamento di codice malevolo (defacement o backdoor).
2. **DELETE Abilitato:** Questo metodo consente l'eliminazione di risorse, un rischio diretto per l'integrità dei dati e la disponibilità del servizio.

Sebbene il test sul path non esistente abbia restituito il corretto 405 (Method Not Allowed) per DELETE e PUT, la risposta positiva su pagine funzionanti (/ e /dvwa/login.php) è sufficiente a classificare la configurazione come **a rischio elevato**.

8 Conclusioni e Raccomandazioni Strategiche

8.1 Sintesi del Progetto

VT Firewall S.r.l. ha completato con successo la fase di progettazione e testing dell'infrastruttura di rete per la Compagnia Theta. Il progetto ha implementato un'architettura di Difesa in Profondità (Defense in Depth) su un edificio di sei piani, garantendo che i 120 host e i dispositivi critici (Firewall, NAS, Web Server, 3 IDS/IPS) siano posizionati strategicamente in una topologia logica isolata e sicura.

8.2 Raccomandazioni di Sicurezza Derivanti dai Test

I test di scansione delle porte e verifica dei verbi HTTP, eseguiti con tool Python sviluppati *ad hoc*, hanno permesso di identificare i seguenti punti critici di configurazione sul Web Server (DVWA di Metasploitable) e richiedono interventi immediati:

1. **Hardening del Web Server:** Si raccomanda di **disabilitare immediatamente** tutti i servizi di rete non essenziali esposti dal server, come FTP (porta 21), Telnet (porta 23) e SSH (porta 22), se non strettamente necessari per la manutenzione. Mantenere questi servizi attivi in un ambiente non affidabile, seppur in DMZ, aumenta drasticamente la superficie di attacco.
2. **Filtraggio dei Verbi HTTP:** Il server esponeva verbi HTTP come PUT o DELETE sul path phpMyAdmin (o altri, se rilevati), creando un rischio critico di manipolazione o distruzione dei contenuti web. È indispensabile configurare il server web (o il Firewall) per **consentire esclusivamente i verbi GET e POST** necessari per il funzionamento dell'applicazione.
3. **Politiche Firewall (ACL):** Le regole del Firewall perimetrale devono essere configurate seguendo il principio del **"Deny All"** di default. Deve essere permesso solo il traffico strettamente necessario, in particolare verso la DMZ, e deve essere bloccato qualsiasi traffico diretto o tentativo di scansione verso la LAN interna.

8.3 Azioni Strategiche e Prossimi Passi

Roadmap per la Sicurezza a Lungo Termine: Manutenzione, Ottimizzazione e Adeguamento

Il presente documento delinea la roadmap strategica per il consolidamento e l'evoluzione continua dell'infrastruttura di sicurezza aziendale. L'obiettivo è garantire una protezione robusta, resiliente e duratura dei dati e dei sistemi informativi, passando da un approccio reattivo a un modello di gestione proattiva del rischio. La strategia si articola su tre pilastri fondamentali: implementazione e verifica, monitoraggio e gestione degli incidenti, e un ciclo continuo di aggiornamento, formazione e conformità.

Fase 1: Implementazione Prioritaria e Verifica delle Contromisure

La fase iniziale del piano prevede l'applicazione immediata e rigorosa delle raccomandazioni di **hardening dei sistemi e delle policy di filtraggio** emerse dalle recenti attività di vulnerability assessment. Questa azione mira a chiudere tempestivamente le vulnerabilità note e a rafforzare le difese perimetrali e interne.

A seguito dell'implementazione di tali contromisure, sarà fondamentale condurre **nuovi cicli di testing e validazione**. Queste verifiche post-implementazione sono cruciali per misurare l'efficacia delle correzioni applicate, confermare la risoluzione delle criticità identificate e assicurare che le modifiche non abbiano introdotto nuove problematiche di sicurezza.

Fase 2: Monitoraggio Continuo e Gestione degli Incidenti

Per garantire una visibilità costante e proattiva sulla postura di sicurezza, si procederà con la valutazione e l'implementazione di una soluzione **SIEM (Security Information and Event Management)**. Questo sistema consentirà la raccolta centralizzata, la correlazione e l'analisi in tempo reale dei log provenienti da tutti gli asset critici, permettendo di identificare pattern anomali e potenziali minacce prima che possano concretizzarsi in incidenti.

Parallelamente, verrà sviluppato e formalizzato un **Piano di Risposta agli Incidenti (IRP)** completo e dettagliato. Tale piano definirà ruoli, responsabilità e procedure operative (playbook) per gestire in modo strutturato ed efficace ogni potenziale violazione o attacco. L'IRP sarà sottoposto a **test periodici** attraverso simulazioni per garantirne la praticabilità e la prontezza operativa del team di risposta.

Fase 3: Aggiornamento, Formazione e Conformità

La resilienza a lungo termine si fonda su una **manutenzione proattiva**. Verrà istituito un programma strutturato di patch management per assicurare l'applicazione tempestiva degli aggiornamenti di sicurezza su tutti i sistemi operativi, le applicazioni e i dispositivi di rete, minimizzando la finestra di esposizione a vulnerabilità note.

Poiché il fattore umano è un anello cruciale della catena di sicurezza, verranno condotte **sessioni di formazione tecnica periodiche** per il personale IT e **campagne di sensibilizzazione (security awareness)** costanti per tutti gli utenti. L'obiettivo è creare una cultura della sicurezza diffusa, in cui ogni dipendente sia consapevole dei rischi e del proprio ruolo nel proteggere le informazioni aziendali.

Infine, per garantire l'allineamento con le best practice di settore e gli obblighi normativi, verranno pianificati **audit di sicurezza regolari**, sia interni che esterni. Queste verifiche sistematiche assicureranno la **conformità a normative e standard** di riferimento (come GDPR e ISO 27001), validando l'adeguatezza dei controlli implementati e identificando aree di miglioramento continuo.

VT Firewall S.r.l. è a completa disposizione della Compagnia Theta per procedere con le fasi esecutive del progetto e garantire un'infrastruttura di rete robusta e protetta contro le minacce cyber.