Introducción a la Ciberseguridad

Adenda Clase 2: Arquitecturas de Seguridad

Teoría: Javier Díaz

jdiaz@unlp.edu.ar

Práctica: Soledad Gomez

Ulises Cabrera

Definicion DMZ

- **Una DMZ (Zona Desmilitarizada) es una subred en una red de computadoras que actúa como una capa de seguridad adicional entre la red interna de una organización (LAN) e Internet**

- **El modelo "DMZ de tres patas“ (DMZ three-legged) es una arquitectura de red que utiliza un solo firewall con tres interfaces de red para segmentar una red en tres zonas de seguridad: Internet (no confiable), la DMZ (zona desmilitarizada) y la red interna (confiable).**

Características de Firewall

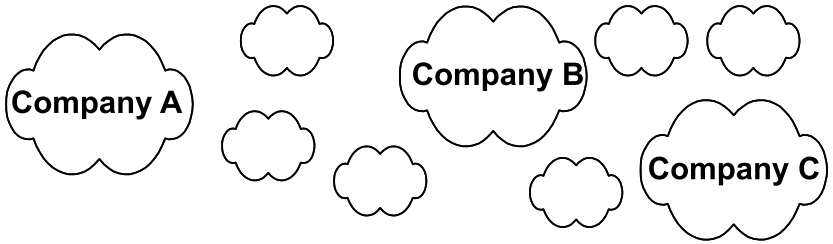
- Reemplazan al SO o son software de base
- Filtros de Paquetes
  - Basados en reglas de Allow/Deny de paquetes
- Extremo de VPN
  - Site-to-site o client-to-site VPNs
- Proxy (o firewall a nivel de aplicación)
  - Chequea contenidos a nivel de la capa de aplicacion (“content filtering”)
  - Realiza funciones de “proxy reverso”
- Administracion
  - Se tra de que exista un Unico punto de adminitracion para todo slos dispositivos Firewall

Características de Firewall (cont.)

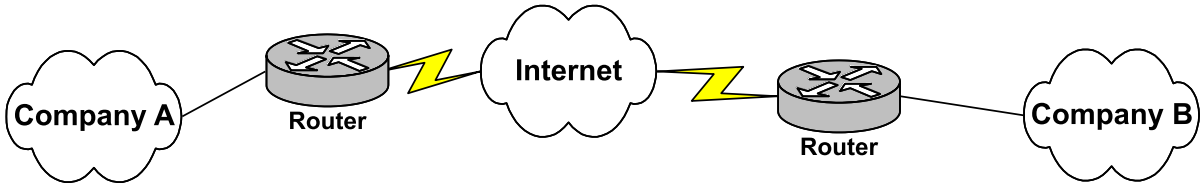
- Funciones Perimetrales
  - Virus scanning
  - Content filtering
- High availability (HA)
  - Debe mantener el estado en condiciones de “failover”
- Balanceo de carga de Firewall (FWLB)
  - HA sumando anchos de banda
- Defensa en profundidad o anidada (Defense-in-depth)
  - Usando múltiples tipes de Firewalls de forma de obtener los beneficios de todos y no ser vulnerables a los puntos débiles de alguno.

Evolucion de Arquitecturas

- Un mundo de islas aisladas (Pre-Internet)

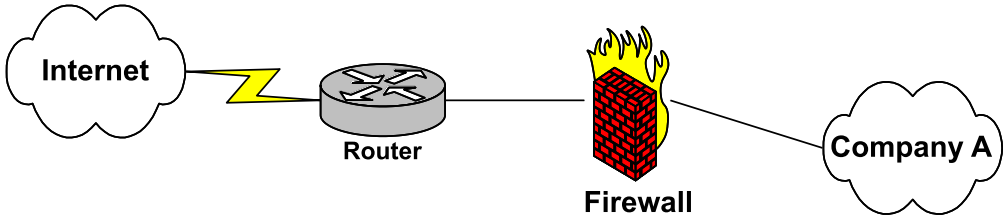


- Yo confío en todos (Inicio de Internet)

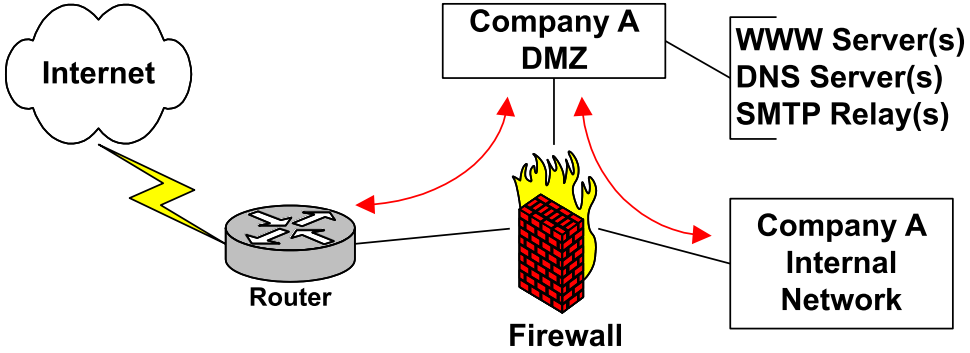


Evolucion de Arquitecturas (cont.)

- Nosotros y Ellos (Filtering & Firewalls)

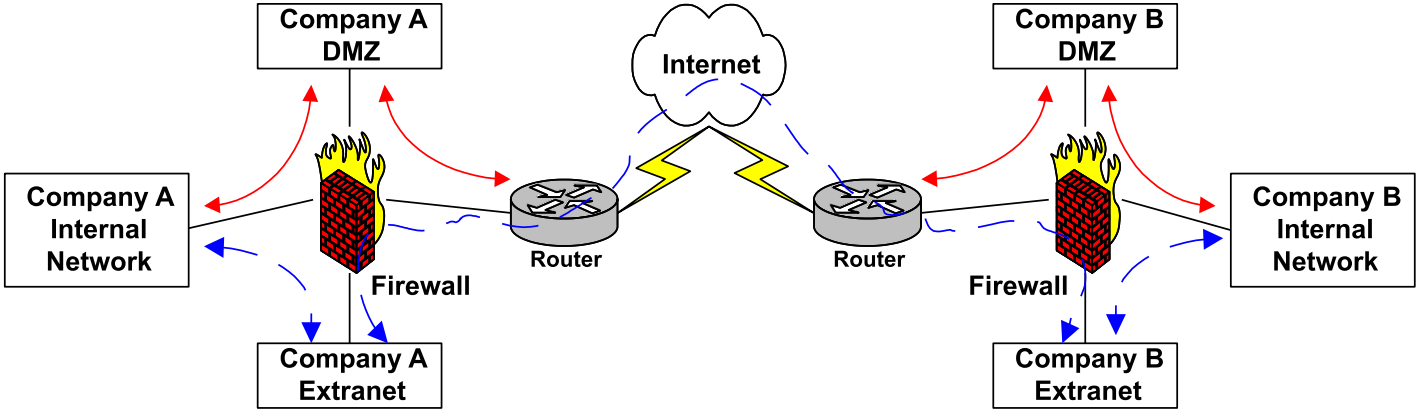


- Comunicación por Internet (DMZs)



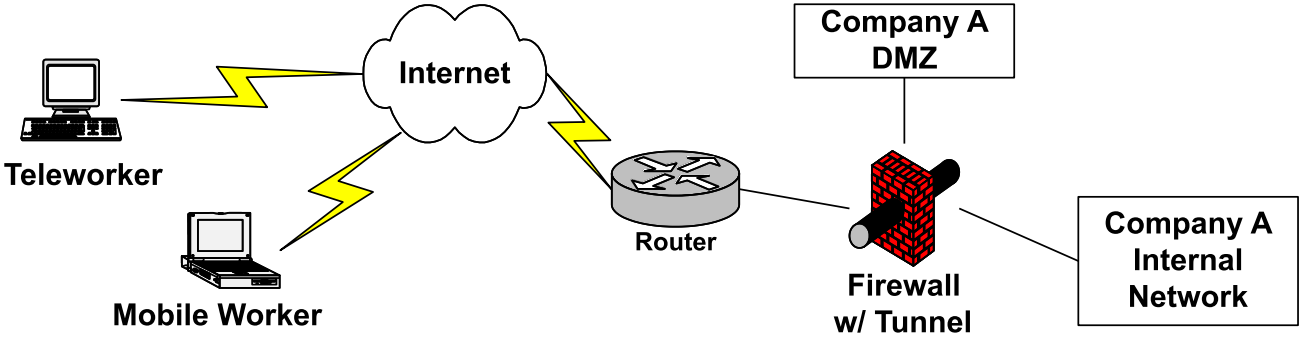
Evolucion de Arquitecturas (cont.)

- Comunidades Virtuales (Extranets & B2B VPNs)
  - –Tuneles VPN Site-to-site
  - –Extranets de negocios para compartir formación



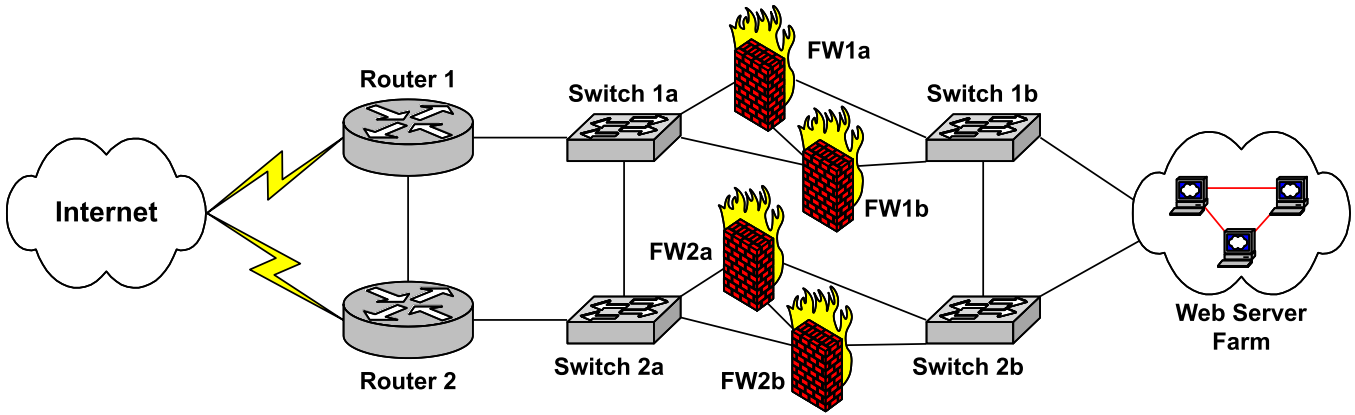
Evolucion de Arquitecturas (cont.)

- Fuerza de trabajo Virtual/Mobil (C2B VPNs)
  - –Mayor ancho de banda para acceder a Internet
  - –Muchos productos de Firewall tienen software cliente de VPN
  - –Concentradores VPN dedicados permiten que conexiones VPN terminen un punto distinto del firewall



Evolucion de Arquitecturas (cont.)

- Ejemplo de FWLB / HA– Centro de Hosting de Web
  - –Objetivos de diseño – Redundancia y ancho de banda importante por suma de links (compuesto)
  - –Sea cuidadose sobre como mantener el estado de la conexión en su diseño



¿Qué es una DMZ?

- Zona Desmilitarizada (DMZ) en redes:
- Espacio intermedio entre la red interna y externa (Internet).
- Contiene servidores públicos: web, correo, DNS.
- Protege la LAN de accesos directos desde Internet.

Tipos de DMZ

- DMZ de Firewall Único (Tres Patas)
- DMZ de Doble Firewall (Subred Protegida)
- DMZ de Doble DMZ

- DMZ de Hardware vs. DMZ de Software

Redes Privadas No Ruteables (RFC 1918)

- Rangos reservados para redes internas:
- 10.0.0.0/8
- 172.16.0.0/12
- 192.168.0.0/16
- No pueden circular en Internet, requieren NAT para salir.

Uso de NAT en DMZ

- El NAT permite publicar servicios de la DMZ:
- Traduce direcciones IP privadas en IP públicas.
- Ejemplo: Servidor web en 10.0.0.10 se publica como 200.55.100.10.
- El firewall controla qué puertos y servicios son accesibles.

¿Qué es NAT?

- NAT (Network Address Translation) es un mecanismo que traduce direcciones IP privadas en públicas y viceversa. Permite que múltiples dispositivos de una LAN accedan a Internet usando una sola dirección IP pública.

Funciones principales

- Ahorrar direcciones IPv4 públicas.
- Mejorar la seguridad ocultando direcciones privadas.
- Facilitar la conexión de múltiples dispositivos a Internet.

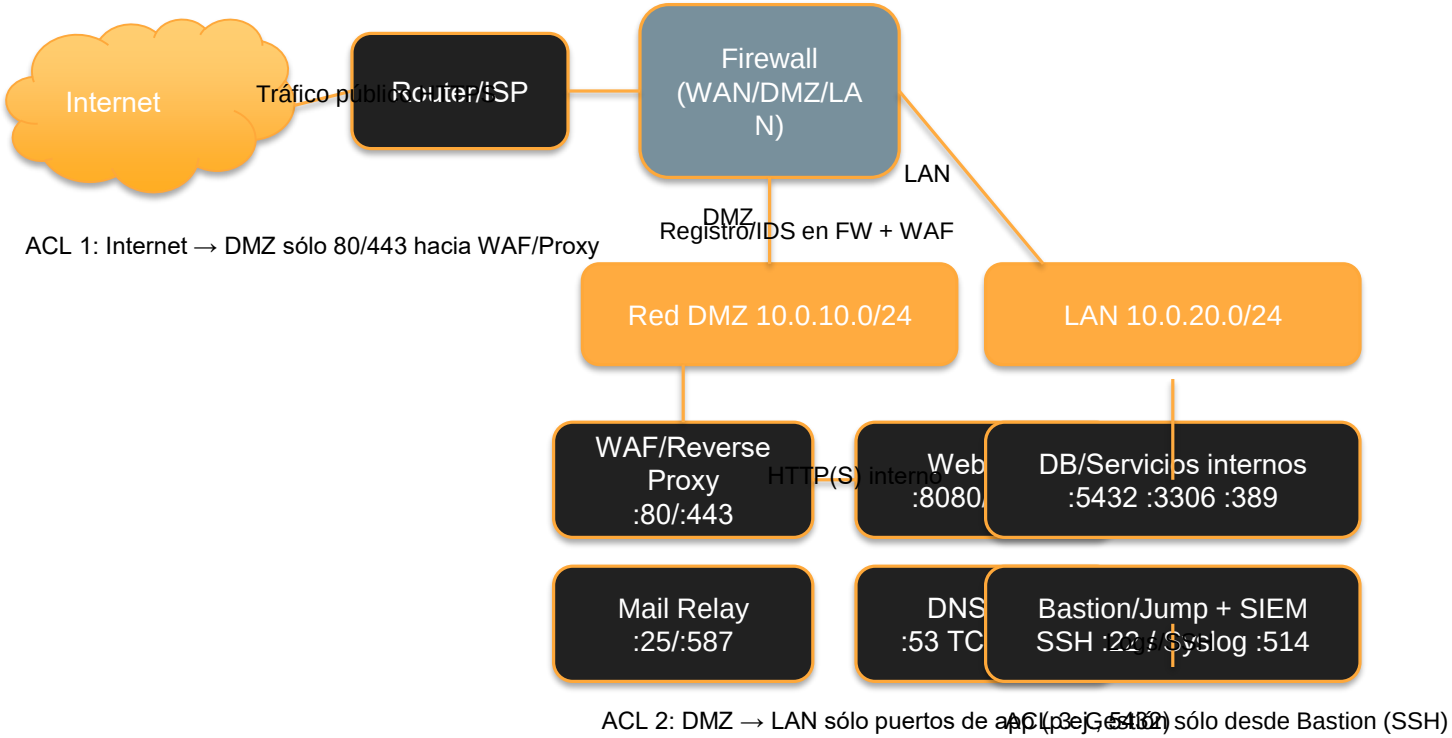
Tipos de NAT

- Static NAT: una dirección privada se asigna a una pública fija.
- Dynamic NAT: se asignan direcciones públicas de un grupo disponible.
- PAT (NAT Overload): varios dispositivos comparten una sola IP pública diferenciados por puertos.

Arquitectura de Seguridad con DMZ — Visión general

- Objetivo: exponer servicios a Internet sin abrir la red interna.
- Topologías: DMZ de 3 patas (three-legged) o con doble firewall (edge + interno).
- Principio: todo el tráfico hacia/desde la DMZ se filtra y registra.
- Regla de oro: Internet ↔ DMZ permitido por puertos específicos; DMZ ↔ LAN sólo lo estrictamente necesario.
- Controles: WAF/Reverse Proxy, autenticación fuerte, IDS/IPS, logging centralizado, parcheo y hardening.

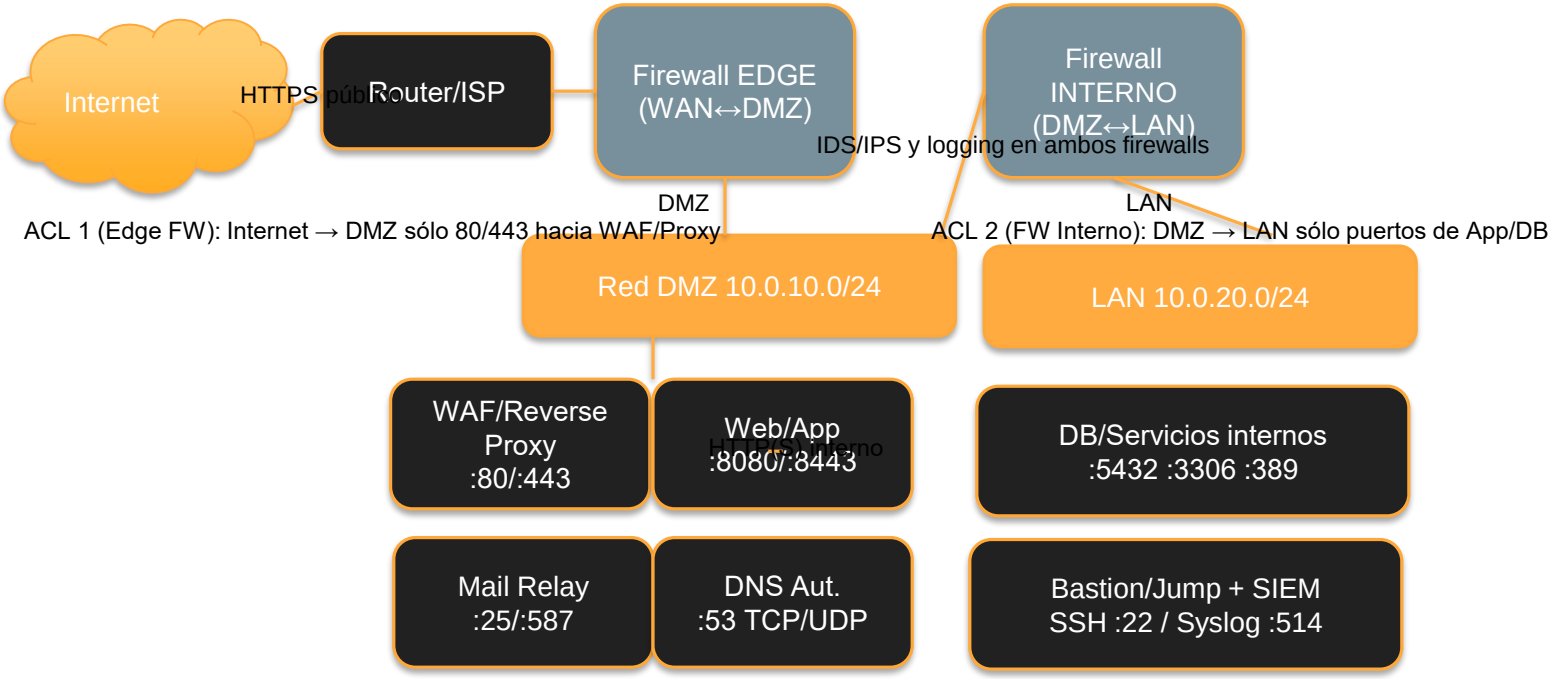
Diagrama — DMZ de 3 patas (three-legged)



Reglas de filtrado y buenas prácticas

- Reglas mínimas:
  - - Internet → DMZ: sólo puertos del servicio (HTTP/HTTPS, SMTP, DNS). Nada directo a LAN.
  - - DMZ → LAN: sólo lo necesario (p.ej., WAF/Proxy → App; App → DB). Denegar por defecto.
  - - Gestión: Bastion/Jump desde LAN a DMZ por SSH; nunca desde Internet.
  - - Salida: DMZ a Internet para actualizaciones vía proxy; bloquear todo lo demás.
- Controles adicionales:
  - - WAF/Reverse Proxy al frente de apps; TLS fuerte; HSTS; rate-limit.
  - - IDS/IPS en DMZ y syslog centralizado; NTP y backups.
  - - Segmentación por VLAN/VRF; mínimo de servicios en servidores DMZ; hardening del SO.
  - - Monitoreo de indicadores: 4xx/5xx inusuales, picos de tráfico, intentos fallidos de login.

Variante — DMZ con doble firewall (Edge + Interno)



Política de BYOD (Bring Your Own Device)

Puntos clave, herramientas open source y arquitectura segura

1. Puntos Salientes de la Política BYOD

- • Definir alcance y dispositivos admitidos.
- • Seguridad de acceso: MFA, RBAC, VPN.
- • Gestión de dispositivos con registro y borrado remoto.
- • Protección de datos: cifrado, contenedores seguros.
- • Cumplimiento normativo (ej. GDPR, Ley 25.326).
- • Concientización y soporte continuo.

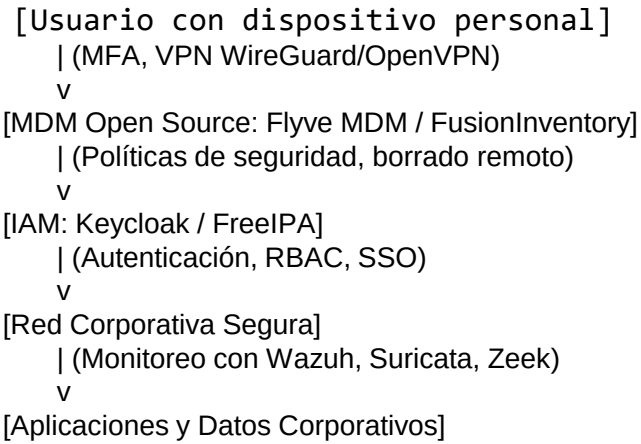
2. Herramientas y Soluciones Open Source

- • **MDM (Mobile Device Management):**
- - Flyve MDM (basado en GLPI).
- - FusionInventory.
- • **IAM (Identity & Access Management):**
- - Keycloak.
- - FreeIPA.
- • **VPN/ZTNA:**
- - OpenVPN, WireGuard.
- - Pritunl (GUI sobre OpenVPN).
- • **EDR/Seguridad de Endpoint**:
- - Wazuh (fork de OSSEC).
- - Suricata + Zeek (detección de anomalías).

Beneficios del Open Source en BYOD

- Transparencia en el código y auditoría comunitaria.
- Flexibilidad para personalizar políticas.
- Reducción de costos de licenciamiento.
- Amplia comunidad de soporte.
- Integración con soluciones propietarias.

Arquitectura Segura de BYOD con Open Source



Políticas para Portátiles (Laptops/Notebooks)

- • Acceso completo a aplicaciones y datos sensibles.
- • Riesgos: copia masiva de datos, malware, robo de credenciales.
- • Controles:
- - Cifrado de disco completo (LUKS, BitLocker).
- - Autenticación multifactor (MFA).
- - VPN obligatoria.
- - EDR/Antimalware (Wazuh, Suricata, Zeek).
- - Parches y actualizaciones obligatorias.

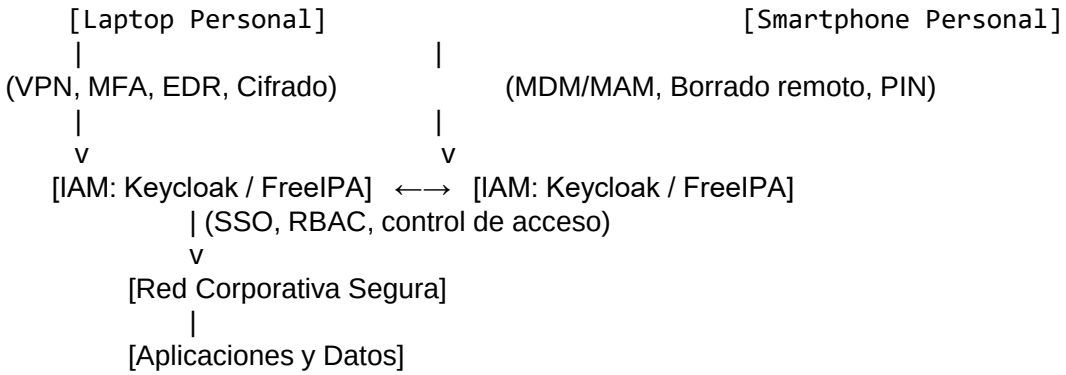
Políticas para Smartphones y Tablets

- Uso principal: correo, mensajería, apps corporativas.
- Riesgos: pérdida/robo, apps no autorizadas, sincronización en nubes personales.
- Controles:
    - Contenedores seguros / MAM.
    - Cifrado de almacenamiento.
    - Borrado remoto vía MDM (Flyve MDM).
    - PIN/Biometría obligatoria.
    - Restricción de apps no autorizadas.

Comparación General: Portátiles vs Smartphones

Aspecto	Laptops/Notebooks	Smartphones/Tablets
Acceso a datos	Alto (documentos, BD)	Medio (apps, correo)
Riesgo principal	Malware, copia masiva de info	Pérdida física, apps no autorizadas
Controles clave	VPN, cifrado de disco, EDR	MDM/MAM, cifrado, borrado remoto
Gestión	Agente en SO	App/Contenedor seguro

Diagrama de BYOD Seguro: Laptops vs Smartphones



Vulnerabilidades en IoT

- Ámbito: dispositivos inteligentes: cámaras IP, wearables, domótica, autos conectados, sensores médicos.

- Impacto típico:
  - Invasión de la privacidad (ej. cámaras hackeadas).
  - Uso de dispositivos como botnets para ataques DDoS.
  - Riesgos en seguridad física (ej. autos conectados, dispositivos médicos).
- Ejemplos:
  - Botnet Mirai (2016): millones de dispositivos IoT usados en ataques masivos.
  - Vulnerabilidades en marcapasos y bombas de insulina reportadas por la FDA.

Seguridad en IoT

- **Objetivos principales:**
  - **Autenticación segura y gestión de credenciales.**
  - **Cifrado de datos en tránsito y en reposo.**
  - **Actualizaciones seguras de firmware/software.**
  - **Resiliencia frente a ataques de botnets y DD**

LoRaWAN Network Architecture:

- **End devices:** dispositivos y sensores que colectan y transmiten data
- **Gateways:** Dispositivos que reciben data de end-devices y los reenvian al network server.
- **Network Server:** Administra la red, maneja el enrutamiento de datos y garantiza la integridad de los mensajes.
- **Application Server:** Almacena y procesas los datos recibidos para la aplicacion.
- **Join Server:** Gestiona el procedimiento de unión y almacena claves de red.

LoRaWAN Network Architecture…

End devices, nodos LoRa

- Qué son: Sensores o actuadores IoT (ej. medidores de agua, sensores de temperatura, trackers GPS).
- Funciones:
  - Generan datos (mediciones, eventos).
  - Usan LoRa PHY para transmitir en bandas ISM.
  - Tienen una AppKey preconfigurada para autenticarse en la red.
- Tipos de activación:
  - OTAA (Over The Air Activation): más seguro, usa Join Server.
  - ABP (Activation by Personalization): más simple, menos seguro.

LoRaWAN Network Architecture…

Gateways

- **Qué son:** Puentes entre los end devices y la red IP.
- **Funciones:**
  - **Reciben las transmisiones LoRa de los dispositivos (uplink).**
  - **Reenvían esos paquetes al Network Server mediante IP (normalmente por MQTT/UDP).**
  - **En downlink, transmiten comandos o configuraciones desde el servidor a los nodos.**

No procesan ni interpretan datos, solo los retransmiten.

LoRaWAN Network Architecture…

Network Server (NS)

- **Rol central en la red:** el “cerebro” que coordina los Gateways y valida la comunicación.
- **Funciones:**
  - Filtra duplicados de paquetes recibidos por múltiples gateways.
  - Verifica integridad con la NwkSKey.
  - Aplica ADR (Adaptive Data Rate) para optimizar ancho de banda y batería.
  - Reenvía mensajes válidos al Application Server.

LoRaWAN Network Architecture…

Application Server (AS)

- Qué hace: el destino final de la información.
- Funciones:
  - Recibe los datos de usuario desde el NS.
  - Usa la AppSKey para desencriptar el payload.
  - Entrega los datos a sistemas de negocio (dashboards, bases de datos, APIs).
  - Permite enviar comandos o configuraciones a los dispositivos (downlink).

LoRaWAN Network Architecture…

Join Server (JS)

- **Especializado en seguridad y activación.**
- **Funciones:**
  - **Maneja las claves maestras (AppKey).**
  - **Durante el proceso OTAA genera claves de sesión:**
    - **NwkSKey → autenticidad e integridad en red.**
    - **AppSKey → confidencialidad del payload.**
  - **Autoriza a los dispositivos a unirse a la red**

IoT LoRaWAN

Seguridad en Dispositivos

- Secure Elements (AppKey/NwkKey): Microchip ATECC608A-TNG(LORA), STSAFE-A110, NXP SE050.

- Firmware seguro: MCUboot (boot firmado), mbedTLS/TinyCrypt, TrustedFirmware-M (ARMv8-M).

- Prácticas: OTAA, LoRaWAN 1.1, claves únicas, rejoin para rotación, FUOTA con firma.

IoT LoRaWAN

Seguridad en Gateways (edge)

- Semtech LoRa Basics Station (TLS/WebSocket) + CUPS para provisión y rotación de certificados.
- Modelos: Kerlink, Tektelic, MultiTech, Laird, MikroTik (ver compatibilidad con Basics Station).
- Hardening: SO actualizado, SSH con claves, iptables/ufw mínimo, postura con Wazuh u osquery/Fleet.

Referencias:

- Semtech LoRa Basics™ Station & CUPS
- LoRa Alliance – LoRaWAN 1.1 Especificación
- MCUboot / mbedTLS / TrustedFirmware-M

Core LoRaWAN y Gestión de Claves

Network/Join/Application Server:

- OSS: ChirpStack (NS/AS + Prometheus), The Things Stack (OSS/Cloud, JS integrado).
- Comerciales: Actility ThingPark, Loriot.

Gestión de claves/certificados:

- PKI: Smallstep step-ca, HashiCorp Vault (PKI + secretos).
- HSM/KMS: YubiHSM 2, Cloud KMS (AWS/Azure/GCP).

Core LoRaWAN y Gestión de Claves

Buenas prácticas de clave:

- Separar AppSKey (en App Server) de claves de red (FNwkSIntKey/SNwkSIntKey/NwkSEncKey).
- Forzar TLS/mTLS entre gateway ↔ LNS/JS ↔ App; vigilar counters y DevNonce.

Referencias:

- ChirpStack docs
- The Things Stack docs
- HashiCorp Vault / Smallstep step-ca

LoRaWAN Network Architecture:

- **End devices:** dispositivos y sensores que colectan y transmiten data
- **Gateways:** Dispositivos que reciben data de end-devices y los reenvian al network server.
- **Network Server:** Administra la red, maneja el enrutamiento de datos y garantiza la integridad de los mensajes.
- **Application Server:** Almacena y procesas los datos recibidos para la aplicacion.
- **Join Server:** Gestiona el procedimiento de unión y almacena claves de red.

LoRaWAN Network Architecture…

End devices, nodos LoRa

- Qué son: Sensores o actuadores IoT (ej. medidores de agua, sensores de temperatura, trackers GPS).
- Funciones:
  - Generan datos (mediciones, eventos).
  - Usan LoRa PHY para transmitir en bandas ISM.
  - Tienen una AppKey preconfigurada para autenticarse en la red.
- Tipos de activación:
  - OTAA (Over The Air Activation): más seguro, usa Join Server.
  - ABP (Activation by Personalization): más simple, menos seguro.

LoRaWAN Network Architecture…

Gateways

- **Qué son:** Puentes entre los end devices y la red IP**.**
- **Funciones:**
  - **Reciben las transmisiones LoRa de los dispositivos (uplink).**
  - **Reenvían esos paquetes al Network Server mediante IP (normalmente por MQTT/UDP).**
  - **En downlink, transmiten comandos o configuraciones desde el servidor a los nodos.**

No procesan ni interpretan datos, solo los retransmiten.

LoRaWAN Network Architecture…

Network Server (NS)

- **Rol central en la red:** el “cerebro” que coordina los Gateways y valida la comunicación.
- **Funciones:**
  - Filtra duplicados de paquetes recibidos por múltiples gateways.
  - Verifica integridad con la NwkSKey.
  - Aplica ADR (Adaptive Data Rate) para optimizar ancho de banda y batería.
  - Reenvía mensajes válidos al Application Server.

LoRaWAN Network Architecture…

Application Server (AS)

- **Qué hace: el destino final de la información.**
- **Funciones:**
  - **Recibe los datos de usuario desde el NS.**
  - **Usa la AppSKey para desencriptar el payload.**
  - **Entrega los datos a sistemas de negocio (dashboards, bases de datos, APIs).**
  - **Permite enviar comandos o configuraciones a los dispositivos (downlink).**

LoRaWAN Network Architecture:

- **Device Authentication and Activation:**

two main methods for devices to join the network:

  - **Over-The-Air-Activation (OTAA): A secure, key exchange-based process.**
  - **Activation-By-Personalization (ABP): A faster but less secure method where keys are pre-provisioned.**

LoRaWAN Network Architecture:

- **Device Authentication and Activation:**

two main methods for devices to join the network:

  - **Over-The-Air-Activation (OTAA): A secure, key exchange-based process.**
  - **Activation-By-Personalization (ABP): A faster but less secure method where keys are pre-provisioned.**

LoRaWAN Network Architecture:

- **Data Security: how LoRaWAN secures data using cryptographic techniques:**
  - **AES Encryption: Used to encrypt the application payload.**
  - **Message Integrity Code (MIC) / Message Authentication Code (MAC): Used to protect against data manipulation.**
  - **Session Keys: Dynamically generated keys for secure communication between device and network.**

LoRaWAN Network Architecture:

Amenazas y Mitigacion:

- Replay Attacks:
- Impersonation:
- Physical Tampering:

LoRaWAN: Replay Attack

- **Riesgo: Captura y retransmisión de tramas legítimas (uplink/downlink) por atacante pasivo y reenvío posterior**
  - **Impactos: re-ejecución de comandos, lecturas falsas de sensores, congestión y daño operativo.**
  - **Señales de alerta: payloads idénticos repetidos, , tráfico atípico.**
- **Mitigacion**
  - **Usar OTAA preferentemente; evitar ABP.**
  - **Registrar hashes/timestamps de tramas recientes para detectar retransmisiones.**
  - **Rotación periódica de claves y forzar rejoin (key refresh).**
  - **Monitoreo: alertas por repeticiones exactas y patrones anómalos.**

LoRaWAN: Impersonation (Suplantación)

- **Riesgo: suplantación de dispositivo o gateway (falsificación de identidad).**
  - **envío de tramas con DevEUI/DeviceAddr falsos o rogue gateways que reenvían paquetes.**
  - **ingreso de datos falsos, exfiltración si claves comprometidas, interrupción del servicio.**
- **Mitigacion**
  - **Provisionar con OTAA y claves en Secure Element; evitar claves compartidas.**
  - **Autenticación TLS/MQTT entre gateways y backend; usar packet forwarder con auth.**
  - **Lista blanca de GatewayEUI y geolocalización correlacionada (RSSI/SNR).**
  - **Implementar device-fingerprinting**

LoRaWAN: Physical Tampering

- **Riesgo: acceso físico al dispositivo para extraer claves o modificar firmware**
  - apertura de carcasa, acceso a puertos JTAG/SWD, microprobing, extracción de memoria
  - compromiso total de dispositivo, suplantación masiva, persistencia (firmware malicioso).
- **Mitigacion**
  - Usar Secure Elements (p.ej. ATECCx08) para almacenar claves y contadores.
  - Habilitar Secure Boot y firma de firmware; deshabilitar JTAG/SWD en producción.
  - Implementar tamper-evident casings, switches tamper y mecanismos de borrado seguro.
  - Claves únicas por dispositivo y no almacenarlas en texto plano en flash.

Arquitectura On-Premises IoT LoRaWAN

Componentes principales:

- End-nodes: Secure Element (ATECC608A/STSAFE/SE050), firmware firmado (MCUboot), OTAA y LoRaWAN 1.1.
- Gateways: LoRa Basics Station (WebSocket TLS) + CUPS para provisión/rotación de certificados.
- Core: ChirpStack (NS/AS) + Join Server propio; gestión de secretos con Vault y PKI con step-ca.

Seguridad de comunicaciones:

- mTLS en GW ↔ LNS/JS y AS ↔ apps; AppSKey solo en Application Server; separación de claves (1.1).
- Rejoin para rotación de claves; alertas por reuse de nonce y reseteo de frame counters.

Arquitectura On-Premises IoT LoRaWAN…

Diseño de red y hardening:

- Segmentación: VLAN/VRF para mgmt de gateways y red del core; firewall solo a puertos LNS/CUPS.
- Gateways protegidos: SSH con llaves, mínimos servicios, postura con Wazuh u osquery/Fleet.

Observabilidad y respuesta:

- Prometheus + Grafana (métricas), Loki/Elastic (logs); paneles para join rate, MIC fails, gateways mudos.
- SIEM con Wazuh; correlación (nonce reuse + múltiples joins) y alertas.

Operación (playbook):

- Alta de dispositivo → provisión de claves/SE → alta en JS/NS → pruebas de join/uplink.
- FUOTA firmado; backup/DR de JS/DB y rotación periódica de certificados.

Arquitectura On-Premises IoT LoRaWAN…

Escalabilidad:

- Balanceo de NS/AS con HAProxy/NGINX; métricas/logs en TSDB/objeto; particionar decoders.

Referencias:

- LoRa Alliance — Especificación LoRaWAN 1.1
- Semtech — LoRa Basics™ Station & CUPS
- ChirpStack — Documentación (NS/AS/Join Server + métricas)
- HashiCorp Vault / Smallstep step-ca — PKI/secretos
- Prometheus/Grafana/Loki — Observabilidad
- Wazuh — SIEM/IDS
- FUOTA — Paquetes LoRa Alliance

Framework de Seguridad para IoT

- **NIST IR 8259 (EE.UU.) 2020** Define un IoT Device Cybersecurity Capability Core Baseline, requisitos para autenticación, actualizaciones, cifrado, logging.
- **ISO/IEC 27400:2022** provee directrices de gestión de riesgos de ciberseguridad en IoT, cubriendo dispositivos, gateways y plataformas en la nube.
- **ISO/IEC 30141:2018** Arquitectura de referencia IoT.2018
- **ETSI EN 303 645 (Europa)** Establece requisitos básicos de ciberseguridad para dispositivos IoT de consumo
- **OWASP IoT Project. Top 10 → Principales riesgos en IoT. 2021**
- **ENISA Guidelines (EU)** Guías específicas para smart homes, eHealth, smart cars, smart infrastructures.

Frameworks de Interoperabilidad IoT

- **AIOTI (Alliance for Internet of Things Innovation)** Promueve estándares abiertos y recomendaciones para la interoperabilidad de IoT en la UE.
- **oneM2M** Estándar global con fuerte adopción europea para arquitectura de interoperabilidad IoT. Define Common Service Functions incluyendo Security Functions (autenticación, autorización, cifrado extremo a extremo).
- **FIWARE (iniciativa UE)** Framework open source de interoperabilidad IoT y smart cities. Integra el NGSI standard API y módulos de IdM (Identity Management) y ciberseguridad para garantizar confianza entre plataformas.
- **ETSI SmartM2M** Complemento europeo de oneM2M, promueve semántica común, APIs abiertas y directrices de seguridad aplicadas en interoperabilidad.

Relación Seguridad IoT y Interoperabilidad IoT

- **La interoperabilidad no puede existir sin seguridad,** porque si los sistemas IoT no gestionan adecuadamente identidad, autenticación y cifrado, no es posible intercambiar datos confiables entre plataformas.
- **ETSI EN 303 645 (seguridad) se complementa con ETSI SmartM2M y oneM2M (interoperabilidad)** → seguridad integrada como requisito transversal.
- **FIWARE y AIOTI** entienden interoperabilidad como interoperabilidad segura, aplicando principios de privacy by design y security by design.
- **Los frameworks de seguridad como NISTIR 8259 y ISO/IEC 27400** sirven de base para asegurar que plataformas interoperables internacionalmente.