Seguretat Infraestructura Sprint 2

VIA – Vilanova Intelligent Airport

Index

Introducció

Aquest informe especifica els requeriments de seguretat aplicables a la infraestructura del projecte VIA – Vilanova Intelligent Airport (PTIN 19-2020). Queda obert a millores i suggerències per part del client empresarial, client tècnic, assessor metodològic i qualsevol altre membre dels equips del terminal 1.

Actual: sprint 2 (20/04/2020)
Anteriors: -
Actualitzacions: -

Servidors

Separació de servidors

Mantenir diferents serveis separats. El servidor DNS ha de ser dedicat.

Afegir serveis extras al Domain Controller porten inconvenients addicionals. Poden ser resolts però és més senzill i donarà menys problemes mantenir separats aquests servidors.
Un error en la seguretat del servei DHCP significaria una reacció en cadena on el servidor DNS passaria a ser vulnerable també. A més no fa falta un error de seguretat per a que ens doni problemes, un concepte tan simple com l'ús de recursos, com pot ser l'ús elevat de CPU o memòria, pot significar que els usuaris no puguin accedir o que fer sol·licituds DNS sigui molt lent.

Servidor DNS

Actualització de versió

Utilitzarem sempre la versió més actual de BIND, amb aquesta sabem que les vulnerabilitats i bugs coneguts estan corregits.

Ocultació de versió

Per tal de donar la menor quantitat d'informació possible ocultarem l'informació de la nostra versió de BIND.

Restricció de transferència de zones DNS

Un tipus d'atac del nostre servidor DNS pot ser fer un intent d'una transferència d'una zona DNS.
Per evitar això hem de limitar quins servidors DNS o quines adreces IP poden fer aquesta transferència.

Deshabilitar la recursivitat DNS

Aquesta recursivitat DNS fa que el servidor que ha rebut la petició, tracta de trobar una resposta a aquesta, anirà preguntant a altres servidors per contestar a aquesta petició. Després es respon a la petició amb les respostes de tots els servidors.
Aquesta opció ve per defecte activada, això deixa el servidor vulnerable a atacs DDoS i atacs d'envenenament DNS.

Servidor DHCP

Tolerància a fallades

Per tal d'evitar que la caiguda del servidor DHCP signifiqui una parada total d'aquest servei necessitem un altre servidor de backup. Els dos servidors estan connectats entre ells i comparteixen la informació, si el servidor principal cau el segon servidor el substitueix.

Control de logs

D'acord amb les recomanacions del client modificarem la configuració DHCP per organitzar els logs del sistema i mantenir un registre. Així podem tenir sempre constancia d'aquests logs i es poden revisar en cas d'un atac.

Adreces IP

Excluirem un rang d'IPs de la *pool* d'adreces DHCP per assignar adreces estàtiques. Si afegim nous dispositius que necessitin una adreça IP estàtica utilitzarem una adreça d'aquest rang prèviament reservat. Així assegurem tenir espai per a nous dispositius que han d'utilitzar una IP estàtica.

Subnetting

Volem segmentar la xarxa per tenir un millor control d'aquesta.
Dividirem la xarxa segons la funció que tinguin els dispositius en el nostre sistema i el nivell de seguretat que vulguem. Per exemple, separarem la xarxa dels servidors de la xarxa de la resta d'usuaris, la xarxa dels usuaris del aeroport de la dels treballadors.

Temps de lloguer DHCP

Una mala gestió d'aquest *lease time* pot significar un esgotament d'adreces IP traduint-se en una mala connectivitat o una falta total d'aquesta.
Aquest temps depèn del subnetting. Un cop decidit la segmentació de la xarxa i la funció que li donarem a cadascuna de les divisions es decidirà el temps que volem per cada subnet.

DHCP snooping

Amb aquesta funció de seguretat tractem d'evitar el *DHCP spoofing* on un atacant es pot fer passar per un dels nostres servidors DHCP i així podria fer atacs del tipus *man-in-the-middle* o *DoS*.
L'objectiu és determinar quins són els missatges que provenen de servidors de confiança.

Docker

Control de versions

Com a la resta dels nostres sistemes volem mantenir el nostre Docker a la versió més actual.

Control d'usuaris:

Ens assegurarem de que únicament usuaris de confiança son membres del grup amb accés Docker i poden accedir a la seva configuració. A més, tots els arxius i directoris han d'estar restringits per a que només els usuaris apropiats tinguin accés.

Privilegis

Per defecte, els contenidors estan autoritzats per adquirir nous privilegis. Això ens deixa vulnerables a atacs del tipus *Privilege escalation*, resultant en que una aplicació tingui més privilegis dels desitjats i pugui realitzar accions no autoritzades per al tipus de privilegi que hauria de tenir realment.
Evitem que els contenidors puguin adquirir nous privilegis canviant aquesta opció per defecte. També eliminarem els permisos SETUID i SETGID dels arxius binaris.

Exclusivitat

A més de tenir servidors dedicats, hem d'implementar a cada contenidor només el software necessari. Amb el mateix motiu, com més software innecessari tinguem al nostre contenidor més alta es la probabilitat de tenir vulnerabilitats en el nostre sistema. Apliquem el mateix concepte amb els directoris i fitxers.

Monitorització

Tractarem de mantenir sempre el nostre sistema sota supervisió. Mantenint els nostres contenidors autentificats i verificats. Assegurem que les imatges del docker que utilitzem no han estat manipulades. A més hem d'eliminar les imatges que no utilitzem.

Connexió

No utilitzem ports privilegiats (per sota de 1024).
No compartim informació entre els contenidor per mantenir-los aïllats com els namespace del host. A menys que sigui totalment necessari.

Root

Un cop actius, els contenidors no haurien de requerir canvis al sistema d'arxius del root. Els canvis que es facin en aquest sistema d'arxius segurament serà amb l'objectiu d'atacar el nostre sistema. El que farem és establir el sistema d'arxius de root a READ-ONLY.

Firewall

Per implementar manualment un firewall amb docker hem de tenir en consideració que aquest tipus d'implementació funciona bé si no volem escalar el sistema amb molta freqüència.

- Primer hem de desactivar les iptables en el docker. Així ens assegurem que docker no sobreescriu les nostres regles.
- Hem de guardar contínuament les regles que creem.
- Afegim les regles que volem aplicar.
- Carreguem aquestes regles.
- Permetem manualment la comunicació entre contenidors.