Mojím zadaním bolo vytvoriť:

„Podniková sieť VLANy v prevedení 2 SSID wi-fi (zamestnanci, hostia) a 4 káblové siete(servers, zamestnanci, management, kamery) , pritom zamestnanci, management musia byť oddelene ale majú mať prístup k serverom a sieť management aj do siete kamery s využitím VLANov“

Vybral som si to ako náhodnú úlohu ktorá sa mi ale nakoniec zapáčila i keď som narazil na veľa nevyžiadaných problémov.

A diagram of a network

Description automatically generated

Najprv som si spravil diagram toho čo som robil, po krátkom vyhľadávaní som zistil že sa takéjto topológii hovorí „Router on a stick“, podľa tohto názvu som neskôr aj vyhľadával dokumentácie. Najprv som si vytvoril vo switchy vlany ktoré budem používať, boli to:

VLAN 10: Employees (Wi=Fi)

VLAN 20: Management

VLAN 30: Servers

VLAN 40: Cameras

VLAN 50: Guests (Wi-Fi)

Tieto vlany som potom nalinkoval na fastEthernet kable

(Fa0/1 – VLAN 10, Fa0/2 – VLAN 20,…)

Tieto káble som pripojil následovne:

Fa0/1 – Access Point-PT (AP pre zamestnancov)

Fa0/2 – počítač (pre manažment) – IP 192.168.20.10

Fa0/3 – server – IP 192.168.30.10

Fa0/4 – počítač (reprezentuje) – IP 192.168.40.10

Fa0/5 – Access Point-PT (AP pre hostí)

Na AP som priradil SSID „zamestnanci“ a pre bezpečnosť pridal autentifikáciu WPA2-PSK a heslo (pass phrase) Employee, vytvoril som si notebook ktoremu som vymenil modul na WMP300N vďaka ktorému sa s heslom vedel pripojiť na AP. Nakoniec som mu ešte pridal IP 192.168.10.10

Podobný proces som zrealizoval pre hostí len bez autentifikácie a IP adresy (pretože s ip adresou hostí pracovať nemôžme), SSID som pomenoval „guest\_wifi“.

Po tomto som switch pripojil k routeru ktorý cez Fa0/7 ktorý je nastavený na trunk a priradil som mu všetky VLANy.

Následovne som vytvoril na routery 5 sub-interfaceov pre GigabitEthernet0/0 (GigabitEthernet0/0.10, GigabitEthernet0/0.20, .....) kde každý reprezentoval jeden VLAN.

Commandy ktoré som používal v CLI boli:

Router(config)# interface GigabitEthernet0/0.30

Router(config-subif)# encapsulation dot1Q 30

Router(config-subif)# ip address 192.168.30.1 255.255.255.0

Ako možno vidieť z commandov, VLANy alebo teda zariadenia v nich detekuje podľa ich IP adresy.

Novovytvoreným sub-interfacom som následovne vytvoril ACL (acces control list)

Router# show access-lists

Extended IP access list 100

10 permit ip 192.168.10.0 0.0.0.255 192.168.30.0 0.0.0.255

30 permit ip 192.168.20.0 0.0.0.255 192.168.30.0 0.0.0.255

40 permit ip 192.168.20.0 0.0.0.255 192.168.40.0 0.0.0.255

50 permit ip 192.168.30.0 0.0.0.255 192.168.10.0 0.0.0.255

60 permit ip 192.168.30.0 0.0.0.255 192.168.20.0 0.0.0.255

70 permit ip 192.168.40.0 0.0.0.255 192.168.20.0 0.0.0.255

Týmto listom povoľujeme prenos dát medzi VLANmi ktorým to povoliť chceme, (je potrebné to spraviť obojstranne preto 3 logické povolenia potrebujú 3 povolenia v ACL)

Router automaticky blokuje všetky prenosy ktoré nie sú v ACL takže toto je všetko čo potrebujeme a naše VLANy sú hotové.

Potencionálne riziká:

1. **Neoprávnený prístup k VLANom**  
   Ak by sa niekto dostal k switchu alebo routeru, mohol by získať kontrolu nad celou sieťou. Preto je dôležité, aby prístup do konfigurácie zariadení bol obmedzený len na autorizovaných používateľov (heslá, autentifikácia). ACL (Access Control List) som navrhol tak, aby blokoval všetky spojenia medzi VLANmi, ktoré nemajú byť povolené. Avšak, ak by sa útočník dostal do siete a poznal pravidlá ACL, mohol by ich obísť – napríklad falšovaním IP adries. Tu by pomohlo ešte nasadenie ďalších opatrení, ako napríklad port security alebo kontrola MAC adries.
2. **Kamery a management VLAN**  
   VLAN kamery sú pripojené na management VLAN, čo má svoje výhody, ale aj riziká. Ak by sa niekto nabúral do kamier, mohol by potenciálne získať prístup aj do management VLAN, čo by bol veľký problém. Tu by sa zišlo ešte viac obmedziť ACL pravidlá na minimálnu potrebnú komunikáciu.