**L2 (KLASIČNI) SWITCHING**

Klasičan svičing obavlja se na nivou L2. Za prosleđivanje paketa koristi se fizička tj. MAC adresa. Gotovo svi današnji svičevi se lako instaliraju i sposobni su da samostalno nauče i mapiraju MAC adrese mrežnih uređaja na odgovarajući fizički port odnosno interfejs. Broj interfejsa zavisi od modela i karakteristika sviča. Učenje se odvija korišćenjem specijalnih tabela koje su zbog brzine najčešće hardverski implementirane (Application Specific Integrated Circuits ili skraćeno ASIC). Ove tabele se sreću pod nazivom CAM odnosno MAC tabele (Cisco koristi CAM terminologiju).



Različiti modeli svičeva

Kada paket stigne na određeni port, posmatra se izvorna i odredišna fizička (MAC) adresa. Izvorna adresa se koristi za učenje o uređajima na mreži, dok se odredišna koristi za prosleđivanje paketa. U tabeli se unosi zapis koji sadrži izvornu fizičku adresu i broj porta na koji je pristigao paket sa izvornom adresom. Odredišna adresa se koristi za određivanje porta na koji treba proslediti paket. U tabeli se zatim vrši pretraga koja adresa kom portu odgovara. Ako se pronađe mapiranje paket se prosleđuje na odgovarajući port. Na ovaj način svičevi prosleđuju saobraćaj samo na odgovarajući port, za razliku od hub uređaja koji prosleđuju na svim portovima osim na port sa kog je stigao paket. Jedini slučaj kada će svič proslediti paket na sve portove je kada proces učenja nije završen, drugim rečima kada svič nema apsolutno nikakvu informaciju o odredišnoj adresi.

Na sledećem primeru može se videti kako klasičan svičing zaista radi: Host X i Y pripadaju segmentu A, dok host Q i host Z pripadaju segmentima B i C respektivno. U segmentu A nalazi se jedan hub uređaj na kome su mrežnim kablom povezani host X i host Y. Pretpostavimo da smo ovu mrežnu arhitekturu upravo obogatili jednim svičem i na njega mrežnim kablom povezali hub, hostove Q i Z.


Prikazaćemo tri najčešća primera komunikacije:

**Host X šalje paket hostu Z**.

Paket stiže do hub uređaja. Hub u skladu sa svojim načinom rada, prosleđuje paket na svim portovima osim na portu gde je stigao paket. Kako su na istom segmentu, host Y prima taj paket, ali ga odbacuje jer analizirajući MAC adresu zaključuje da paket nije namenjen njemu. U isto vreme paket se prosleđuje i sviču. Svič analizira izvornu adresu, i u svojoj MAC tabeli koja je trenutno prazna, unosi prvi zapis u obliku <MAC address, port, timestamp>. Na ovaj način svič zna na kom se interfejsu nalazi host X (port Fa0/1). Ovo predstavlja proces učenja (eng. learning). Da bi prosledio paket svič mora da analizira odredišnu adresu paketa. Kako trenutno svič zna samo za host X, paket se prosleđuje na svim interfejsima (eng. flooding), osim na interfejs sa kog je stigao. Paket primaju Q i Z, ali potvrđuje jedino host Z. Kada potvrdni paket stigne do sviča, uređaj saznaje na kom je segmentu host Z i dodaje novi zapis u MAC tabeli. Analizirajući tabelu svič sada zna na kom je portu host X i prosleđuje paket na taj port (nema flooding-a). Ovo prosleđivanje paketa je glavna funkcija svič uređaja (eng. forwarding). Bitno je napomenuti da se zbog hardverske implementacije ovo prosleđivanje paketa odvija veoma brzo. Dalji saobraćaj sa hosta X ka hostu Z se odvija brzo i jednostavno bez flooding-a, koristeći samo forwarding.


**Host Y šalje paket hostu X**.

Kada paket stigne do hub uređaja, ovaj prosleđuje kao i u prvom primeru na sve portove. Kako su hostovi X i  Y na istom mrežnom segmentu, paket stiže do hosta X i time je jednosmerna komunikacija završena. Međutim, svič to ne zna, jer i on u istom vremenskom trenutku dobija paket namenjen hostu X. Svič sada uči na kom se interfejsu nalazi host Y. Analizirajući odredišnu adresu i zapise u tabeli za hostove X i Y, na osnovu istog interfejsa (Fa0/1) zaključuje da su X i Y na istom segmentu. Zbog toga, primljeni paket se odbacuje i nema prosleđivanja od strane sviča, tj. nema dupliranja paketa. Ovaj proces se naziva filtriranje (eng. filtering).


**Host X šalje paket hostu Z koji je zamenjen drugim uređajem ili više nije aktivan.**

Ako ste se pitali koja je svrha polja timestamp u MAC tabeli, evo odgovora. Baš zbog ovakvih i sličnih promena na mreži, polje timestamp se koristi kako bi MAC tabela uvek bila ažurna. Uvek kada svič analizira izvornu adresu dolaznog paketa, ažurira se timestamp polje trenutnim vremenom ili kreira novi zapis ukoliko ne postoji u tabeli. Svič vremenom prolazi kroz sve zapise u tabeli i proverava timestamp polje. Ako uređaj na mreži ne procesira pakete u određenom vremenskom intervalu onda se zapis za taj uređaj, u MAC tabeli posmatra kao nevažeći i uklanja iz tabele. Najčešće se ova tehnika ažuriranja, u engleskoj terminologiji sreće pod nazivom aging, a interval posle kog se zapisi odbacuju kao zastareli se naziva mac-table-aging-time. Ovaj termin se sreće kod programibilnih svičeva. Još jedan bitan razlog zašto se koristi ova tehnika je ograničena memorija sviča i broj zapisa u tabeli. Na ovaj način memorija sviča se optimizira. Ako host X pokuša da komunicira sa hostom Z koji nije na mreži, tj. zapis koji mapira adresu hosta Z u odgovarajući port je izbrisan iz tabele, onda se radi klasičan flooding kao u prvom primeru. Na primljeni paket niko neće odgovoriti tako da će X posle nekog vremena zaključiti da uređaj više ne postoji na mreži.


Što se tiče tehnika za prosleđivanje paketa, u svičevima mogu da se nađu dve najpoznatije: Cut-Through i Store-and-Forward. Kod prve tehnike, zbog brzine, paket se prosleđuje odmah nakon analiziranja odredišne adrese, dok se kod druge paket analizira u celini i prosleđuje jedino ako nema grešaka. Generalno gledano, korišćenjem prve tehnike dobija se na brzini, dok korišćenjem druge tehnike smanjuje se kolizija i propagiranje paketa s greškom se ovde zaustavlja. Danas je razlika između ove dve tehnike u vremenu prosleđivanja postala zanemarljiva, tako da se Store-and-Forward tehnika nameće kao bolje rešenje.