

hakin9

Ataki na drugą warstwę modelu OSI

Alfredo Andrés, David Barroso

Artykuł opublikowany w numerze 5/2005 magazynu *hakin9*. Zapraszamy do lektury całego magazynu.

Wszystkie prawa zastrzeżone. Bezpłatne kopiowanie i rozpowszechnianie artykułu dozwolone
pod warunkiem zachowania jego obecnej formy i treści.

Magazyn *hakin9*, Software-Wydawnictwo, ul. Piaskowa 3, 01-067 Warszawa, pl@hakin9.org



Pod lupą

Ataki na drugą warstwę modelu OSI

Alfredo Andrés, David Barroso

stopień trudności



Druga warstwa modelu OSI to najsłabsze ogniwo bezpieczeństwa sieciowego każdego systemu. Realizowanie ataków na tę warstwę jest rzadkie, jednak udany atak na nią może być tak samo niebezpieczny, jak każdy inny.

Warstwa łącza danych jest jednym z najsłabiej zabezpieczonych elementów sieci. Administrator często spina ze sobą switchy (przełączniki) i konfiguruje je, po czym przestaje zaprzętać sobie nimi głowę. Testy penetracyjne często wykrywają przełączniki, które korzystają z podatnych na ataki wersji IOS i nie są w żaden sposób zabezpieczone. Dodatkowo pokutuje myślenie, że wdrożenie VLAN w sieci powstrzymuje napastników. Prawda jest inna – taka architektura również może zostać pokonana, a jeżeli do tego dojdzie, możliwe będą ataki na wyższe warstwy OSI (sniffing hasel, *man-in-the-middle*).

Pocieszeniem może być to, że pakiety warstwy łącza danych nie podróżują przez sieci IP – na przykład przez Internet – w związku z czym ataki są ograniczone do sieci wewnętrznych. Statystyki wskazują jednak, że ataki lokalne mogą być tak samo niebezpieczne, jak te z zewnątrz. Trzeba pamiętać, że gdy intruz z zewnątrz pokona firewall i dostanie się do strefy zdemilitaryzowanej (DMZ), to ataki na warstwę drugą mogą mu pozwolić na ucieczkę z tej strefy i w konsekwencji na uzyskanie dostępu do

całej naszej sieci. Przyjrzyjmy się więc nierzalczym aspektom warstwy łącza danych, temu, jak może ją wykorzystać atakujący oraz temu, w jaki sposób możemy chronić swój sprzęt. Wszystkie przykłady odnoszą się do urządzeń Cisco, ale część z nich może działać ze sprzętem innych producentów.

Większość danych i obserwacji autorzy zebrali podczas badań i tworzenia narzędzia

Z artykułu dowiesz się...

- jakie są specyfikacje protokołów warstwy drugiej OSI: STP, CDP, DTP, IEEE 802.1Q, VTP,
- jak przeprowadzać ataki na te protokoły,
- jak obronić system przed tymi atakami,
- jak używać użytecznego dla administratorów i testerów bezpieczeństwa narzędzia Yersinia.

Co powinieneś wiedzieć...

- powinieneś znać podstawy warstwy drugiej OSI,
- powinieneś mieć wiedzę o technologiach Cisco.

Siedem warstw OSI

W 1977 r. zaproponowano model o nazwie *Open Systems Interconnection* (OSI), którego celem było opracowanie standardu zgodności sprzętu różnych producentów. Model ten definiuje kilka warstw przesyłu danych, od najniższej (warstwa fizyczna) do najwyższej (warstwa aplikacji). Są one silnie zależne od siebie, a ich nagłówki są zwykle dodawane przy przechodzeniu z warstwy niższej do wyższej. Tych siedem warstw to:

- warstwa 1 – warstwa fizyczna – obsługuje komunikację (i nadzór) w kanale sieciowym,
- warstwa 2 – warstwa łącza danych – ustala metody dostarczania bloków danych,
- warstwa 3 – warstwa sieciowa – odpowiada za rutowanie pakietów danych,
- warstwa 4 – warstwa transportowa – odpowiada za niezawodną (bezbłędną) transmisję danych,
- warstwa 5 – warstwa sesji – umożliwia kontrolę nad dialogiem między aplikacjami,
- warstwa 6 – warstwa prezentacji – pomaga aplikacjom ustalić format danych, co czyni prezentację uporządkowaną,
- warstwa 7 – warstwa aplikacji – ustanawia metody umożliwiające aplikacjom dostęp do modelu OSI (czyli do sieci).

Yersinia. Czasem odnalezienie dodatkowych informacji czy publicznie dostępnego kodu było niemożliwe,

więc obserwacje oparto na analizie behawioralnej, a nie na opublikowanych standardach.

Narzędzie Yersinia

Do przeprowadzania opisywanych ataków na warstwę drugą będziemy używać narzędzia Yersinia, napisanego przez autorów niniejszego artykułu. Yersinia jest przenośna (napisano ją w C, z wykorzystaniem bibliotek libpcap i libnet) i wielowątkowa (obsługuje wielu użytkowników i wiele równoczesnych ataków). Może być używana do analizy, edycji i obserwacji pakietów sieciowych, pozwala też zapisywać ruch sieciowy w formacie *pcap*.

Najnowsza (0.5.5.1) wersja programu Yersinia obsługuje następujące protokoły:

- *Spanning Tree Protocol* (STP),
- *Cisco Discovery Protocol* (CDP),
- *Dynamic Trunking Protocol* (DTP),
- *Dynamic Host Configuration Protocol* (DHCP),
- *Hot Standby Router Protocol* (HSRP),
- IEEE 802.1Q,
- *Inter-Switch Link Protocol* (ISL),
- *VLAN Trunking Protocol* (VTP).

Yersinia działa w jednym z trzech głównych trybów:

- linia poleceń (CLI) może być używana do doraźnych ataków – ten tryb dodano, by ułatwić testerom stosowanie narzędzia w skryptach,
- demon sieciowy pozwala używać Yersinii zdalnie – CLI jest bardzo podobne do stosowanego w produktach Cisco,
- graficzny interfejs (GUI) napisany w ncurses.

Wszystkie opisywane ataki były przeprowadzane w trybie GUI, chociaż wykorzystanie innych trybów nie byłoby problemem. Aby poznać możliwości narzędzia, należy nacisnąć `[h]` podczas pracy w trybie GUI (`yersinia -I`). Uwaga: tryb ten wymaga dużej liczby wierszy i kolumn, jeśli więc GUI nie zadziała, warto ponowić próbę po maksymalizacji okna terminala.

Yersinia umożliwia także przeprowadzanie innych ataków – na przykład HSRP, DHCP – my jednak skoncentrujemy się tylko na tych, które dotyczą warstwy drugiej. Nazwa narzędzia pochodzi od bakterii, która wywołała epidemię Czarnej Śmierci w średniowiecznej Europie – *Yersinia pestis*.

STP (Spanning Tree Protocol)

Celem protokołu STP jest uniknięcie pętli sieciowych podczas łączenia segmentów sieci. Może istnieć tylko jedna unikalna ścieżka łącząca dwa urządzenia. Każdy pakiet STP nosi nazwę BPDU (*Bridge Protocol Data Unit*). Możemy go zidentyfikować patrząc na jego format: pakiet IEEE 802.3 z nagłówkiem IEEE 802.2 i docelowym adresem MAC 01:80:C2:00:00:00 (patrz Rysunek 1).

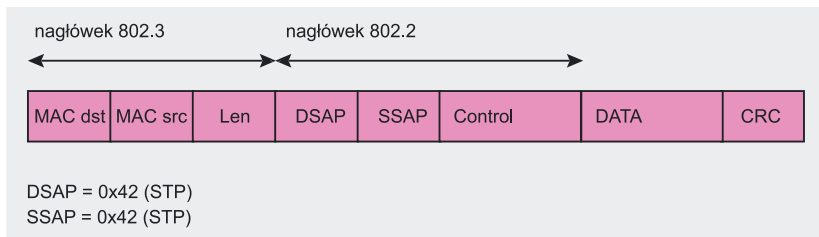
Istnieją dwa typy pakietów BPDU: *Configuration* (konfiguracja) i *Topology Change Notification* (TCN – powiadomienie o zmianie topologii). Pierwszy jest wysyłany co pewien czas i pokazuje konfigurację sieci, drugi jest natomiast wysyłany za każdym razem, kiedy wykryta zostanie zmiana w sieci (port otwarty lub zamknięty). Więcej informacji o STP można znaleźć w standardzie IEEE 802.1D (patrz Ramka *W Sieci*).

Ataki

Największą słabością STP jest brak uwierzytelniania i nadzoru. Każde urządzenie i każda osoba, w tym atakujący, może wysłać BPDU i uczestniczyć w transmisji protokołowej.

Aby zrozumieć ataki, trzeba poznać format *Configuration* (patrz Rysunek 2):

- *PID* (2 bajty): protokół, zawsze wartość zerowa,
- *Version* (1 bajt): wersja STP, może to być 0 (STP), 1 (RSTP) lub 3 (MSTP),
- *Message type* (1 bajt): typ BPDU: *Configuration* (0x00) lub TCN (0x80),
- *Flags* (1 bajt): kilka ustawień portów (przydatne dla RSTP) i bit powiadamiania o zmianie topologii,
- *Root ID* (8 bajtów): ID nadrzędnego urządzenia,
- *Root path cost* (4 bajty): koszt przebycia trasy do urządzenia nadrzędnego,
- *Bridge ID* (8 bajtów): ID nadawcy BPDU,



Rysunek 1. Budowa pakietu BPDU



Rysunek 2. Struktura pakietu BPDU typu Configuration

- **Port ID** (2 bajty): numer portu (IEEE lub Cisco STP BPDU), z którego wysyłany jest pakiet BPDU,
- **Message age** (2 bajty): czas, który upłynął od wysłania obecnej konfiguracji przez urządzenie nadrzędne,
- **Maximum age** (2 bajty): kiedy aktualna wiadomość o konfiguracji powinna być skasowana,
- **Hello time** (2 bajty): czas pomiędzy wysłaniem pakietów konfiguracyjnych BPDU,
- **Forward delay** (2 bajty): czas, który powinny odczekać mostki przed przejściem w nowy stan, po zmianie topologii.

i obliczenie trasy między urządzeniami uczestniczącymi w rozgałęziającym się drzewie (*spanning tree*). Na początku wszystkie biorą udział w wyborze urządzenia nadrzędnego – wybrane zostaje to z najniższym ID – a potem wszystkie trasy są obliczane przy każdej zmianie w sieci. Nowe urządzenie nadrzędne jest wybierane, gdy obecne znika z sieci lub gdy pojawia się nowe, z ID niższym niż ID obecnego urządzenia nadrzędnego.

Do dzieła

Przyjrzyjmy się trzem możliwym atakom na STP. Dwa z nich to ataki typu *Denial of Service* (DoS), wymuszające na wszystkich urządzeniach uczestniczących w STP ciągle prze-

liczanie tras. Powoduje to niestabilność sieci, ponieważ każdy switch obciąża podczas przeliczania zarówno procesor, jak i pamięć. Ataki takie mogą również powodować powstawanie pętli sieciowych. W najgorszym przypadku cała sieć przestanie działać – wszędzie będą krążyć zduplikowane pakiety, zapychając sieć i powodując ogólną awarię.

Dekodowanie pakietów

Chociaż jednym z zastosowań Yersinii jest dekodowanie i obserwacja pakietów wartwy drugiej, można do tego celu użyć innych analizatorów protokołów, takich jak tcpdump czy Ethereal. Jeśli na przykład chcemy przechwytać pakiety STP, Ethereal może być uruchomiony w następujący sposób:

```
# ethereal -f stp
```

liczanie tras. Powoduje to niestabilność sieci, ponieważ każdy switch obciąża podczas przeliczania zarówno procesor, jak i pamięć. Ataki takie mogą również powodować powstawanie pętli sieciowych. W najgorszym przypadku cała sieć przestanie działać – wszędzie będą krążyć zduplikowane pakiety, zapychając sieć i powodując ogólną awarię.

Takie ataki są raczej proste. Polegają na wysłaniu tysięcy pakietów BPDU (w pierwszym przypadku *Configuration*, a w drugim *TCN*) z losowo wygenerowanymi źródłowymi adresami MAC (oraz, w przypadku *Configuration*, innymi polami, na przykład *Bridge ID*). Jest to symulacja tysięcy nowych urządzeń podłączających się do sieci i zgłaszających chęć uczestniczenia w STP – nic dziwnego, że efektem jest chaos.

Oba ataki można przeprowadzić za pomocą narzędzia Yersinia. Nazywają się one *sending conf BPDUs* i *sending tcn BPDUs* (klawisz [x] w trybie GUI). Listingi 1 i 2 pokazują reakcję przełącznika na te ataki.

Trzeci atak polega na próbie uzyskania nadrzędnej roli STP. W tym celu najpierw zostaje przechwycony pakiet BPDU zawierający nadrzędne ID, a następnie atakujący system jest konfigurowany tak, że zachowuje się jak inne urządzenie sieciowe chcące uczestniczyć w STP – ale mające ID niższe niż aktualne urządzenie

Listing 1. Rezultaty ataku DoS (wysyłanie pakietów Configuration BPDU)

```
01:20:26: STP: VLAN0001 heard root 32768-d1bf.6d60.097b on Fa0/8
01:20:26: STP: VLAN0001 heard root 32768-9ac6.0f72.7118 on Fa0/8
01:20:26: STP: VLAN0001 heard root 32768-85a3.3662.43dc on Fa0/8
01:20:26: STP: VLAN0001 heard root 32768-3d84.bc1c.918e on Fa0/8
01:20:26: STP: VLAN0001 heard root 32768-b2e2.1a12.dbb4 on Fa0/8
```

Listing 2. Rezultaty ataku DoS (wysyłanie pakietów BPDU TCN)

```
01:35:39: STP: VLAN0001 Topology Change rcvd on Fa0/8
01:35:39: STP: VLAN0001 Topology Change rcvd on Fa0/8
01:35:39: STP: VLAN0001 Topology Change rcvd on Fa0/8
01:35:39: STP: VLAN0001 Topology Change rcvd on Fa0/8
01:35:39: STP: VLAN0001 Topology Change rcvd on Fa0/8
```

Listing 3. Rezultaty ataku Claiming Root Role

```
01:58:48: STP: VLAN0001 heard root 32769-000e.84d4.2280 on Fa0/8
01:58:48:      supersedes 32769-000e.84d5.2280
01:58:48: STP: VLAN0001 new root is 32769, 000e.84d4.2280 on port Fa0/8, cost 19
```

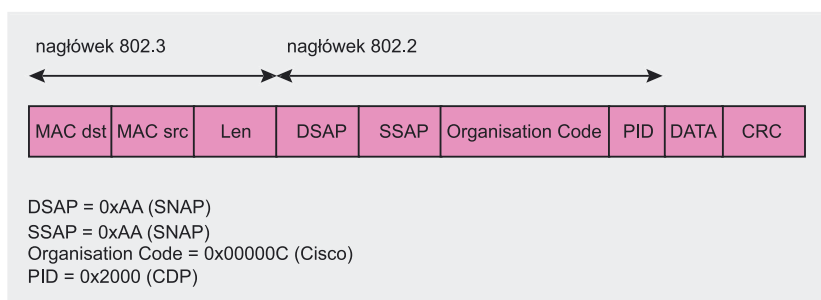
nadrzędne. Falszywy nadrzędny ID jest zmniejszany tylko o 1, więc nie różni się zbyt wiele od prawdziwego – administrator na pierwszy rzut oka nie zauważy zmiany.

Najpoważniejszą konsekwencją takiego ataku jest niestabilność sieci. Trzeba pamiętać, że wszyscy uczestnicy sieci wysyłają pakiety TCN do nadrzędnego urządzenia po wykryciu zmiany. Tylko wtedy nadrzędne urządzenie wysyła pakiet *Configuration BPDU* z bitem zmiany ustawionym na 1 (pole *Flags*), aby nakazać wszystkim uczestnikom przeliczenie tras. Jeśli atak się powiedzie, nowe, fałszywe urządzenie nadrzędne unieważnia pakiety TCN wysyłane przez switche, toteż żaden z nich nie przelicza swoich tras. W efekcie niszczone jest struktura sieci.

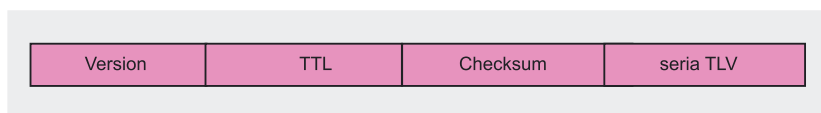
Aby przeprowadzić taki atak za pomocą Yersinii, musimy najpierw nacisnąć [d] w celu wypełnienia pakietu domyślnymi wartościami, a potem uruchomić atak *Claiming Root Role* (najpierw wcisnąć [x], a następnie wybrać atak 4). Składa się on z dwóch etapów: najpierw przechwytyjemy konfiguracyjny pakiet BPDU by poznać nadrzędny ID, a następnie wysyłamy nowy, spreparowany pakiet BPDU *Configuration*, co pewną liczbę sekund określoną polem *hello time*. Reakcję przełącznika wiadać na Listingu 3.

Stary nadrzędny ID miał postać 32769-000e.84d5.2280, zaś nowy to 32769-000e.84d4.2280. Jeśli uważnie się przyjrzymy, zauważymy, że piętnasty znak to 4 zamiast 5. Nasze wirtualne urządzenie ma niższy ID, dzięki czemu zostaje wybrane jako nowy nadrzędny ID protokołu STP.

Są też inne możliwości ataków na STP. Niektóre zaimplementowano w narzędziu Yersinia. Jeden z nich nosi nazwę *Causing Eternal Root Elections* i polega na nieprzerwanym wysyłaniu pakietów z coraz niższymi ID, co powoduje nieskończony wybór urządzenia nadrzędnego i totalny chaos w sieci. Inny zwie się *Claiming Root Role with MiTM* i jest atakiem typu *man in the middle*. Możemy też spróbować *Claiming Other Role*, co oznacza próbę zachowywa-



Rysunek 3. Struktura pakietu CDP



Rysunek 4. Pola pakietu CDP

nia się tak, jak inny, zwykły switch. Jest to tylko przykład możliwego ataku (*proof of concept*), bez żadnych negatywnych konsekwencji.

Obrona

Aby uniknąć ataków STP na urządzenia Cisco, administrator może:

- wyłączyć STP tam, gdzie nie jest potrzebne,
- używać *Spanning Tree Portfast BPDU Guard Enhancement* i *Spanning Tree Protocol Root Guard Enhancement* (patrz Ramka W Sieci).

cje: pakiet IEEE 802.3 z nagłówkiem 802.2 SNAP i docelowym adresem MAC *multicast* w postaci 01:00:0C:CC:CC:CC (patrz Rysunek 3). Pakiet CDP zawsze zawiera ciekawe informacje o właściwościach wysyłającego go urządzenia, na przykład:

- nazwę urządzenia,
- model,
- wersję IOS,
- adres IP (może mieć więcej niż jeden),
- domenę VTP,
- możliwości (switch, ruter, mostek itp.).

Dane te, okresowo wysyłane przez każde urządzenie Cisco, dostarczają informacji cennych przy późniejszych atakach. Domyślnie CDP jest włączony i wysyła te informacje co 180 sekund – czyli co trzy minuty.

Ataki

Wysyłanie i odbieranie pakietów CDP odbywa się bez uwierzytelniania. Dane są przesyłane czystym

CDP (Cisco Discovery Protocol)

CDP to własnościowy protokół Cisco, umożliwiający komunikację między różnymi urządzeniami sieciowymi Cisco. Inni producenci również mogą go używać, jednak dopiero po kupieniu licencji na technologię.

Najprostszym sposobem identyfikacji pakietu CDP jest sprawdzenie, czy posiada on następujące funk-

Tabela 1. Przykłady zestawów TLV

Zawartość TLV	Typ	Długość	Wartość
0001 0008 7a61 7065	Device ID (0x0001)	8 (0x0008) (dwa bajty na typ, dwa bajty na długość, cztery bajty na wartość)	zape (0x7a 0x61 0x70 0x65)
000b 0005 01	Duplex type (0x000b)	5 (0x0005) (dwa bajty na typ, dwa bajty na długość, jeden bajt na wartość)	0x01 (Full Duplex)

**Listing 4. Rezultat ataku CDP DoS**

```
# show cdp neighbours
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone

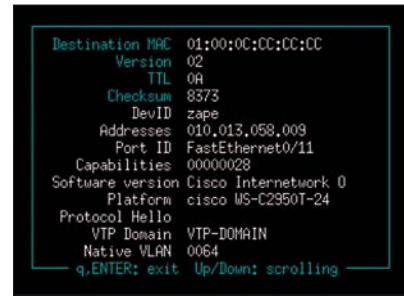
Device ID      Local Intrfce   Holdtme    Capability   Platform   Port ID
2EEEWWW       Gig 0/1         253        yersinia     yersinia   Eth 0
ZCCCUU9       Gig 0/1         250        T S I r      yersinia   Eth 0
J222FFX       Gig 0/1         249        R T          yersinia   Eth 0
WAAASS6       Gig 0/1         240        R B I r      yersinia   Eth 0
2IIWWWE       Gig 0/1         249        T B H I      yersinia   Eth 0
K333FFX       Gig 0/1         234        R T          yersinia   Eth 0
TBBB007       Gig 0/1         252        B H r        yersinia   Eth 0
3KKYKYY       Gig 0/1         250        R B H        yersinia   Eth 0
TBBBPP7       Gig 0/1         252        S H I r      yersinia   Eth 0
```

tekstem, co bardzo ułatwia ataki. W dodatku format CDP jest objaśniony na stronie internetowej Cisco (patrz Ramka *W Sieci*). Pakiet składa się z następujących pól (Rysunek 4):

- **Version** (1 bajt) – wskazuje na wersję CDP, zwykle 1 lub 2,
- **TTL** (1 bajt) – *Time To Live* – “czas życia” pakietu CDP,
- **Checksum** (2 bajty) – określa poprawność pakietu,
- **TLV** (różna długość) – seria *Type, Length, Value*. To pole zawiera właściwe dane, reprezentowane przez listę zestawów TLV, z których każdy ma następujący format: *Type* (2 bajty) – typ da-

nych (na przykład *Device ID, Address, Port ID*), *Length* (2 bajty) – długość TLV oraz *Value* (zmienna długość) – właściwa wartość. Przykłady zestawów TLV znajdują się w Tabeli 1, zaś Rysunek 5 przedstawia Yersinię wyświetlającą TLV przykładowego pakietu.

Znając ten format możemy udawać urządzenie sieciowe, poprzez wysłanie spreparowanego pakietu CDP. Warto też wiedzieć, że stare wersje Cisco IOS są podatne na atak DoS – podatność tę odkrył FX z grupy Phenoelit (patrz Ramka *W Sieci*). Jeśli wyśle się dużo pakietów CDP z różnymi ID (zachowujących się jak



Rysunek 5. TLV przykładowego pakietu oglądanego w Yersinii

różne urządzenia sieciowe), to wyczerpie się pamięć w urządzeniu. W efekcie przestanie ono działać i będzie musiało zostać zrestartowane, by zachowywało się poprawnie. Taki atak może spowodować odłączenie segmentu sieci lub, jeśli celem jest ruter, brak dostępu do Internetu aż do restartu.

Do dzieła

Jeśli jesteśmy połączony z siecią, w której działają urządzenia obsługujące CDP, Yersinia w trybie GUI szybko pokaże tryby CDP tych urządzeń. Pierwszy atak związany z CDP opiera się na wspomnianej wyżej podatności; nie potrzebujemy żadnych dodatkowych danych. W GUI Yersinii, w trybie CDP, wystarczy nacisnąć [x] i wybrać atak *flooding CDP*

Listing 5. Rezultat ataku CDP DoS – log switcha

```
00:06:08: %SYS-2-MALLOCFAIL: Memory allocation of 224 bytes failed from 0x800118D0, alignment 0
Pool: Processor Free: 0 Cause: Not enough free memory
Alternate Pool: I/O Free: 32 Cause: Not enough free memory
-Process= "CDP Protocol", ipl= 0, pid= 26
-Traceback= 801DFC30 801E1DD8 800118D8 80011218 801D932C 801D9318
00:06:08: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:09: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:10: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:11: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:12: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:13: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:14: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:15: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:16: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:17: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:18: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:19: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:20: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:21: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:22: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:23: ../src-calhoun/strata_stats.c at line 137: can't not push event list
00:06:38: %SYS-2-MALLOCFAIL: Memory allocation of 140 bytes failed from 0x801E28BC, alignment 0
Pool: Processor Free: 0 Cause: Not enough free memory
Alternate Pool: I/O Free: 32 Cause: Not enough free memory
```

Co to jest trunking

W telefonii *trunk* (magistrala) jest linią, przez którą można przesyłać między dwoma punktami więcej niż jeden kanał głosu lub danych w tym samym czasie. W znaczeniu sieciowym, magistrala może łączyć dwa switchy. W ten sposób przesyła się ruch wielu sieci VLAN przez jedno i to samo łącze.

table. Rezultaty widoczne są na Listingu 5, zaś log switcha widać na Listingu 5.

Za pomocą Yersinii można także przeprowadzić atak umożliwiający tworzenie wirtualnych urządzeń Cisco. Kiedy administrator sprawdza sieciowych sąsiadów prawdziwych urządzeń, wszystkie spreparowane pojawią się w konsoli Cisco. Ten atak nie ma żadnych negatywnych konsekwencji poza zdenerwowaniem administratora (który oczywiście będzie chciał się dowiedzieć, skąd wzięty się nowe urządzenia podłączone do sieci).

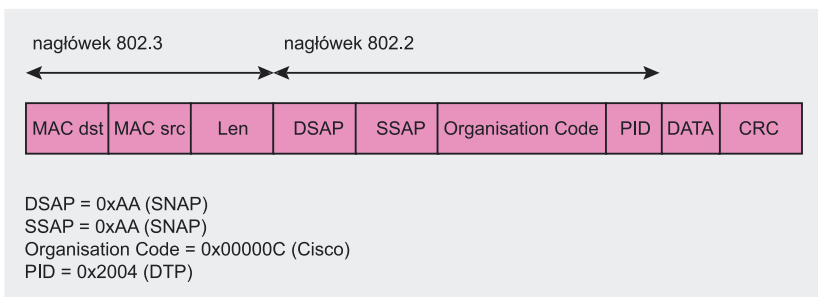
Obrona

Jedyną skuteczną obroną przed atakami CDP jest wyłączenie tego protokołu poleceniem `no cdp run`. Sam protokół nie posiada żadnych rozszerzonych funkcji bezpieczeństwa.

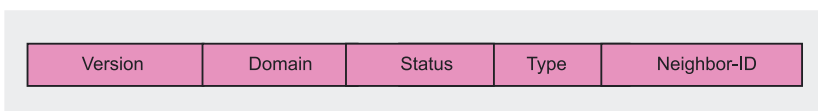
DTP (Dynamic Trunking Protocol)

DTP to własnościowy protokół Cisco, który ustanawia magistrale (ang. *trunk* – patrz Ramka *Co to jest trunking*) między przełącznikami warstwy drugiej. Pakiety DTP zwykle mają wartość `01:00:0C:CC:CC:CC` jako docelowy adres MAC oraz ramkę 802.3 zawierającą nagłówek 802.2 SNAP (patrz Rysunek 6). Protokół ten jest dostępny we wszystkich modelach przełączników Cisco, poza serią XL.

W urządzeniach Cisco protokół DTP jest domyślnie włączony i gotowy do negocjacji z każdym (fizycznym) portem switcha. Aby jednak ustanowić magistralę, trzeba wiedzieć, jak negocjować DTP. Specyfikacja protokołu jest własnością Cisco, co odrobinę utrudnia sprawę; autorzy artykułu byli zmuszeni do



Rysunek 6. Struktura pakietu DTP



Rysunek 7. Budowa pakietu DTP (bez nagłówek Ethernet)

konać inżynierii wstecznej (*reverse engineering*) ruchu między dwoma switchami połączonymi magistralą, aby rozpracować format DTP.

DTP negocjuje aktywację magistrali i typ enkapsulacji używany do przesyłania ruchu przez dany port. Najpopularniejszym rodzajem enkapsulacji jest IEEE 802.1Q, obsługiwana przez większość przełączników Cisco – jej specyfikacja to publiczny standard. W zamian może być jednak używana ISL, będąca z kolei własnościowym protokołem Cisco, dostępnym jedynie w urządzeniach z najwyższej półki. Głównym powodem używania enkapsulacji jest znakowanie pakietów odpowiednimi tagami VLAN. Umożliwia to przełącznikom wysyłanie pakietów w odpowiednie miejsca.

Ataki

Jak już wspomnieliśmy, DTP nie używa uwierzytelniania nadawcy. Jest aktywne domyślnie na wszystkich portach – jedynym warunkiem jest możliwość negocjowania DTP.

Jeśli tak jest, możemy zdobyć dostęp do innych VLAN. Aby wiedzieć jak negocjować DTP, musimy poznać format pakietów DTP (patrz Rysunek 7):

- *Domain* (32 bajty) – łańcuch ASCII identyczny ze skonfigurowaną domeną VTP,
- *Status* (1 bajt) – pokazuje status portu: *on*, *off*, *desirable* lub *auto*; domyślnie *desirable* – możemy zacząć negocjować DTP,
- *Type* (1 bajt) – obsługiwany typ enkapsulacji: *ISL*, *802.1Q*, *negotiated* (ISL lub 802.1Q) lub *native*,
- *Neighbor-ID* (6 bajtów) – identyfikuje urządzenie wysyłające pakiet, zwykle adres MAC portu.

W przypadku urządzeń Cisco pierwszym krokiem przy negocjacji DTP jest wysłanie trzech pakietów – jeden na sekundę – pokazujących status łączenia w magistralę i wymagany typ enkapsulacji. Następnie pakiet DTP jest wysyłany co 30 sekund. Yersinia implementuje to zachowanie jako

Listing 6. Status VLAN przed atakiem

```
zipi# sh vlan
VLAN Name                Status    Ports
-----
1    default                active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                           Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                           Fa0/9, Fa0/14, Fa0/15, Fa0/16
                                           Fa0/17, Fa0/18, Fa0/19, Fa0/24
                                           Gi0/1, Gi0/2
100  Office                 active    Fa0/10, Fa0/11, Fa0/12, Fa0/13
200  Internet              active    Fa0/20, Fa0/21, Fa0/22, Fa0/23
```

**Listing 7. Status portu DTP widziany w konsoli switcha**

```
zipi# sh dtp int Fa0/10
DTP information for FastEthernet0/10:
  TOS/TAS/TNS:                ACCESS/DESIRABLE/ACCESS
  TOT/TAT/TNT:                NATIVE/802.1Q/802.1Q
  Neighbor address 1:         000000000000
  Neighbor address 2:         000000000000
```

Listing 8. Status portu po ataku

```
zipi# sh dtp int fa0/10
DTP information for FastEthernet0/10:
  TOS/TAS/TNS:                TRUNK/DESIRABLE/TRUNK
  TOT/TAT/TNT:                802.1Q/802.1Q/802.1Q
  Neighbor address 1:         666666666666
  Neighbor address 2:         000000000000
```

Listing 9. Porty przypisane do VLAN po ataku

```
zipi# sh vlan
VLAN Name                Status    Ports
-----
1    default                active   Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                         Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                         Fa0/9, Fa0/14, Fa0/15, Fa0/16
                                         Fa0/17, Fa0/18, Fa0/19, Fa0/24
                                         Gi0/1, Gi0/2
100  Office                  active   Fa0/11, Fa0/12, Fa0/13
200  Internet                active   Fa0/20, Fa0/21, Fa0/22, Fa0/23
```

wątek odpowiedzialny za zadanie. Z drugiej strony, do zmiany własnego statusu w razie konieczności niezbędny jest nadzór nad statusem innego urządzenia – Yersinia dokonuje tego za pomocą pętli odbierającej pakiety DTP. Po kilkukrotnym sprawdzeniu, Yersinia zmienia swój status DTP stosownie do statusu innego urządzenia.

Do dzieła

Sprawdźmy jak wygląda atak – użyjemy switcha Catalyst 2950T z systemem IOS 12.1(22) EA3. Nazwa hosta to *zipi*, są tutaj dwie sieci VLAN: *Office* (porty Fa0/10, Fa0/11, Fa0/12 i Fa0/13) oraz *Internet* (porty Fa0/20, Fa0/21, Fa0/22 i Fa0/23). Domena VTP została zmieniona na *Yersinia*.

Wszystkie inne parametry mają wartość domyślną. Na Listingu 6 widać status VLAN przed atakiem.

W trybie GUI wybierzmy ekran protokołu DTP. Jeśli jest on obecny w sieci, zobaczymy dane DTP po niecałych 30 sekundach. Możemy także obejrzeć status portów DTP z konsoli przełącznika: nasz port to Fa0/10, a jego status jest domyślny (patrz Listing 7).

Za pomocą klawisza [d] musimy wypełnić dolne pola okna domyślnymi wartościami. Następnie przy użyciu klawisza [e] modyfikujemy pole *Neighbor-ID*, wprowadzając wartość 666666666666. Aby zakończyć edycję naciskamy [Enter].

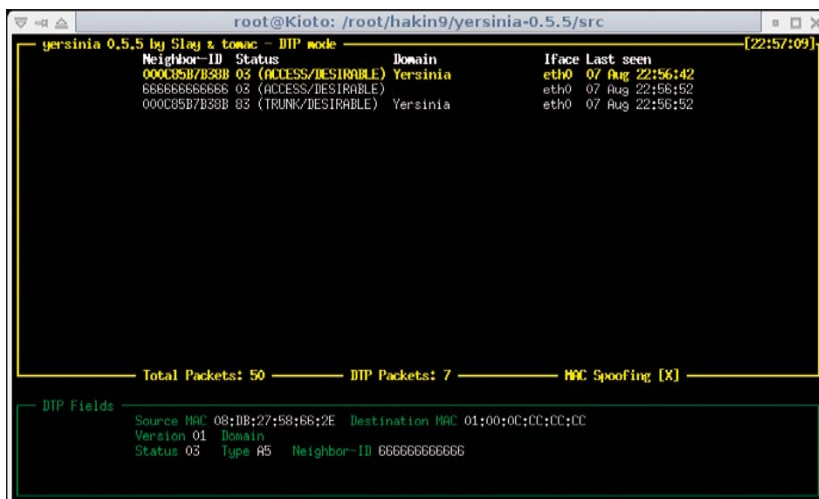
Teraz przełączmy się do okna ataku DTP przy użyciu [x] i wybieramy atak typu *enabling trunking*. Status portu DTP zmieni się na *TRUNKING*, a pole *Neighbor address 1* będzie zawierać nasz ID (patrz Listing 8). Co więcej, jeśli tak jak poprzednio spojrzymy na porty przypisane VLAN, zobaczymy, że naszego portu Fa0/10 nie ma na liście VLAN (patrz Listing 9). W głównym oknie Yersinii zobaczymy nowe pakiety – te zmodyfikowane przez nas mają pole *Neighbor-ID* równe 666666666666 (patrz Rysunek 8). Od tej pory będziemy mogli przeprowadzać ataki na protokoły 802.1Q i VTP. Dodatkowo będziemy też mogli imitować zachowanie innego switcha, co umożliwi śledzenie ruchu VLAN (innego niż ten, do którego jesteśmy podłączeni).

Obrona

Jedyną skuteczną metodą obrony przeciwko atakom DTP jest wyłączenie automatycznego zestawiania magistral za pomocą polecenia `switchport mode access` – administrator będzie wtedy musiał ręcznie włączać zestawianie (w konfiguracji switcha) dla każdej nowej magistrali.

IEEE 802.1Q

Specyfikacja protokołu IEEE 802.1Q jest publiczna. Opisuje ona format wykorzystywany w pakietach przechodzących przez łącza magistralowe. Ze względu na otwarty charakter specyfikacji, ten standard jest obecnie



Rysunek 8. Efekt ataku DTP

akceptowany przez większość producentów i jest popularną metodą zestawiania magistral pomiędzy przełącznikami różnych firm. Nie jest to jednak standard jedyny – większość producentów stosuje własne rozwiązania, na przykład Cisco ma swój zastrzeżony protokół ISL (*Inter-Switch Link*).

Kiedy switch odbiera ramkę, dodaje do niej znacznik 802.1Q (4 bajty), przelicza FCS (*Frame Check Sequence*) i wysyła zmodyfikowaną ramkę do łącza magistralowego. Rysunek 9 przedstawia pola dodane przez 802.1Q do ramki Ethernet_II. Pole VID identyfikuje sieć VLAN, do której należy pakiet, może przyjmować wartości między 0 a 4096. Teoretycznie – jeśli nawiążemy połączenie magistralowe, a switch obsługuje 802.1Q – możemy wysyłać pakiety do różnych VLAN.

Ataki

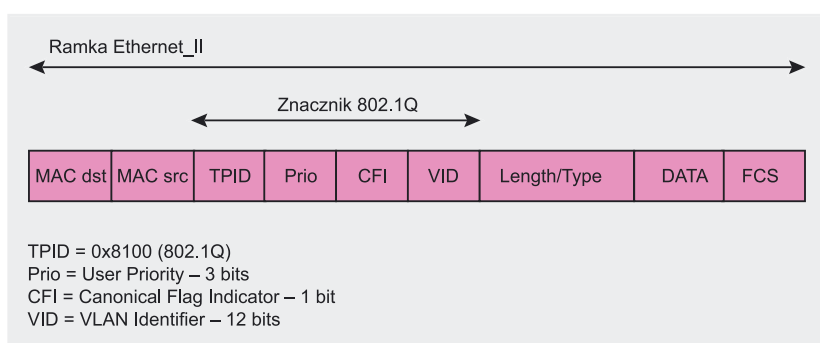
Do używania 802.1Q konieczne jest zestawienie magistrali. W poprzedniej sekcji mogliśmy zobaczyć, jak utworzyć magistralę za pomocą DTP, a dodatkowo jak określić, czy z 802.1Q będzie używana enkapsulacja. Załóżmy więc, że łącze magistralowe już na odpowiednim porcie istnieje.

Ataki przeciwko 802.1Q można podzielić na dwa typy:

- przesyłanie ramek 802.1Q do sieci VLAN nienależącej do atakującego,
- użycie podwójnie enkapsulowanych ramek 802.1Q – ten rodzaj ataku dodaje dwa znaczniki do oryginalnej ramki, w celu użycia sieci VLAN z drugiego znacznika jako docelowej po tym, jak switch usunie pierwszy tag.

Do dzieła

Spróbujmy najpierw wysłać za pomocą Yersinii podwójnie enkapsulowane ramki 802.1Q. Na ekranie 802.1Q wypełniamy pola domyślnymi wartościami (klawisz [d]) i przechodzimy do trybu edytora (klawisz [e]). Teraz zmienimy wartość *Source MAC* na 66:66:66:66:66:66, wartość *VLAN* na 16, a *VLAN2* na 1. Na koniec wyjdźmy z trybu edytora [En-



Rysunek 9. Pola dodane przez 802.1Q do ramki Ethernet_II

Listing 10. Zdekodowany za pomocą Ethereal pakiet Yersinia ICMP Echo Request

```
Ethernet II, Src: 66:66:66:66:66:66, Dst: ff:ff:ff:ff:ff:ff
  Destination: ff:ff:ff:ff:ff:ff (ff:ff:ff:ff:ff:ff)
  Source: 66:66:66:66:66:66 (66:66:66:66:66:66)
  Type: 802.1Q Virtual LAN (0x8100)
802.1q Virtual LAN
  111. .... = Priority: 7
  ...0 .... = CFI: 0
  ... 0000 0001 0000 = ID: 16
  Type: 802.1Q Virtual LAN (0x8100)
802.1q Virtual LAN
  111. .... = Priority: 7
  ...0 .... = CFI: 0
  ... 0000 0000 0001 = ID: 1
  Type: IP (0x0800)
Internet Protocol, Src Addr: 10.0.0.1 (10.0.0.1), ←
  Dst Addr: 255.255.255.255 (255.255.255.255)
  Protocol: ICMP (0x01)
  Source: 10.0.0.1 (10.0.0.1)
  Destination: 255.255.255.255 (255.255.255.255)
Internet Control Message Protocol
  Type: 8 (Echo (ping) request)
  Checksum: 0xb953 (correct)
  Identifier: 0x0042
  Sequence number: 00:42
  Data (8 bytes)
0000 59 45 52 53 49 4e 49 41                                YERSINIA
```

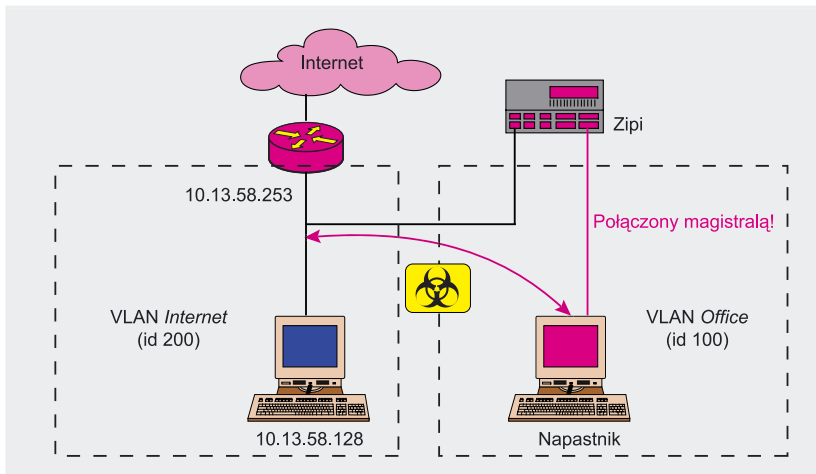
ter]. Teraz przejdźmy to okna ataku klawiszem [x] i wybierzmy atak *sending 802.1Q double enc. packet*.

Yersinia użyje 802.1Q do wysłania pakietów *ICMP Echo Request*

z ładunkiem *YERSINIA*. Listing 10 pokazuje taki pakiet, zdekodowany za pomocą narzędzia Ethereal (część pól usunięto dla większej przejrzystości). Dokładnie widać, że

Listing 11. Konfiguracja VLAN wykorzystana w ataku

```
zipi# sh vlan
VLAN Name                Status    Ports
-----
1    default                active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                           Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                           Fa0/9, Fa0/14, Fa0/15, Fa0/16
                                           Fa0/17, Fa0/18, Fa0/19, Fa0/24
                                           Gi0/1, Gi0/2
100  Office                 active    Fa0/10, Fa0/11, Fa0/12, Fa0/13
200  Internet               active    Fa0/20, Fa0/21, Fa0/22, Fa0/23
```



Rysunek 10. Mapa sieci dla 802.1Q

wysłaliśmy podwójnie enkapsulowaną ramkę 802.1Q – najpierw z VLAN 16, a następnie z VLAN 1.

Ten atak tylko pokazuje, że możemy wstrzyknąć ruch sieciowy do innych VLAN (tak zwany *VLAN-hopping*). Można też przeprowadzać bardziej zaawansowane ataki, na przykład *man-in-the-middle*. Przyszłe wersje Yersinii będą oferować takie metody.

Do następnego ataku będziemy potrzebowali bardziej złożonej konfiguracji. Założmy, że mamy switch o nazwie *zipi* z systemem IOS 12.1(22) EA3 i dwiema sieciami VLAN: *Office* (porty Fa0/10, Fa0/11, Fa0/12 i Fa0/13) oraz *Internet* (Fa0/20, Fa0/21, Fa0/22 i Fa0/23) – patrz Listing 11. Domena VTP została zmieniona na

Yersinia. Inne zmienne konfiguracyjne mają wartość domyślną.

Rysunek 10 przedstawia strukturę sieci. Urządzenie atakującego jest połączone z VLAN 100 (*Office*) przez port Fa0/10. Do VLAN 200 podłączony jest komputer z systemem Windows, o adresie IP 10.13.58.128. Jak widać na Rysunku 10, urządzenia z VLAN *Internet* łączą się z Internetem przez router o adresie IP 10.13.58.253, a VLAN *Office* nie ma dostępu do Internetu.

Chcemy uzyskać dostęp do ruchu generowanego przez komputer z Windows, położonego w sieci VLAN *Internet*. W tym celu użyjemy ataku *sending 802.1Q arp poisoning*. Przed rozpoczęciem ataku powinniśmy jednak upewnić się, że wpis ARP dla

10.13.58.253 (routera internetowego) w tablicy ARP systemu z Windows (10.13.58.128) jest prawdziwym adresem MAC routera. Ponadto, tak, jak w poprzedniej sekcji, musimy włączyć i wynegocjować łącze magistralowe.

Wybermy teraz ekran 802.1Q. Prawdopodobnie zobaczymy pakiety kierowane do adresu MAC *broadcast* lub *multicast*. Ten typ pakietów jest wysyłany do wszystkich portów należących do tej samej sieci VLAN (w naszym przypadku 200, VLAN *Internet*) i dodatkowo do wszystkich portów z wynegocjowaną magistralą (takich, jak nasz port). Wypełnijmy teraz pola domyślnymi wartościami i przejdźmy do trybu edytora, by zmienić pole *Source MAC* na 66:66:66:66:66:66. Następnie przejdźmy do okna ataku i wybierzmy *sending 802.1Q arp poisoning* – potrzebuje on kilku parametrów, więc pojawi się nowe okno z następującymi polami:

- *IP to poison* – adres IP, który chcemy zastąpić, w naszym przypadku 10.13.58.253 (router internetowy),
- *IP VLAN* – identyfikator sieci VLAN, do której chcemy wysłać pakiety, w naszym przypadku 200 (VLAN *Internet*),
- *ARP IP Source* – adres IP, który będzie użyty do odgadnięcia adresu MAC routera internetowego; musi być nieprzypisanym jeszcze adresem w tej samej VLAN, w której znajduje się *IP to poison* – użyjmy 10.13.58.66.

Teraz możemy już rozpocząć atak. Jeśli wszystko pójdzie dobrze, adres MAC 66:66:66:66:66:66 pojawi się w tablicy ARP komputera z Windows (10.13.58.128), a na ekranie Yersinii pojawi się więcej danych (patrz Rysunek 11).

Spróbujmy zrozumieć, jakie czynności wykonuje Yersinia:

- szuka adresu MAC routera internetowego przy użyciu fałszywych (*spoof*) pakietów ARP,
- kiedy adres MAC routera jest już znany, uruchamia się nowy wątek, który wysyła jeden pakiet

```

root@fredy: /usr/local/src/coder/yersinia/src
yersinia 0.5.5.1 by Slay & towar - 802.1Q mode [16:28:14]
VLAN L2Protol Src IP Dst IP IP Prot Iface Last seen
0200 0800 IP 10.13.58.128 66.249.93.104 06 tcp eth0 08 Aug 16:26:40
0200 0108 PVST eth0 08 Aug 16:28:12
0200 0800 IP 10.13.58.128 66.249.93.104 06 tcp eth0 08 Aug 16:26:41
0200 0800 IP 10.13.58.128 66.249.93.104 06 tcp eth0 08 Aug 16:26:41
0200 0800 IP 10.13.58.128 66.249.93.104 06 tcp eth0 08 Aug 16:26:40
0200 0806 ARP 10.13.58.253 eth0 08 Aug 16:28:13
0100 0108 PVST eth0 08 Aug 16:28:13
0200 0800 IP 10.13.58.128 66.249.93.104 06 tcp eth0 08 Aug 16:26:40
0200 0800 IP 10.13.58.128 66.249.93.104 06 tcp eth0 08 Aug 16:26:40
0200 0800 IP 10.13.58.128 66.249.93.104 06 tcp eth0 08 Aug 16:26:41
Total Packets: 1266 802.1Q Packets: 447 MAC Spoofing [X]

802.1Q Fields
Source MAC 66:66:66:66:66:66 Destination MAC FF:FF:FF:FF:FF:FF
VLAN 0001 Priority 07 CFI 00 L2Protol 0800 VLAN2 0002 Priority 07 CFI 00
L2Proto2 0800 Src IP 010.000.000.001 Dst IP 255.255.255.255 IP Prot 01

```

Rysunek 11. Efekt ataku 802.1Q

Co to jest domena VTP

Domena VTP to łańcuch ASCII wspólny dla wszystkich przełączników z tej samej grupy lub jednostki (*entity*). Wszystkie pakiety VTP zawierają ten łańcuch, a w dodatku domena jest obecna w niektórych polach CDP – nawet domena DTP jest tym samym łańcuchem.

ARP reply na sekundę; to zatruje tablicę ARP Windows fałszywym adresem MAC adresu IP 10.13.58.253 (ruter internetowy) – 66:66:66:66:66:66.

Yersinia może teraz przechwycić wszystkie dane wysyłane przez komputer z Windows (10.13.58.128) do Internetu, a następnie przepisać je i odesłać do źródłowego VLAN (VLAN *Internet*, ID 200) jako skierowane do prawdziwego adresu MAC rutera internetowego (10.13.58.253). Jeśli chcemy zapisać dane sieciowe, wystarczy nacisnąć klawisz [s] – zostaną zapisane w formacie pcap. Lepiej, żeby właściciel systemu Windows nie używał nieszyfrowanych haseł.

Obrona

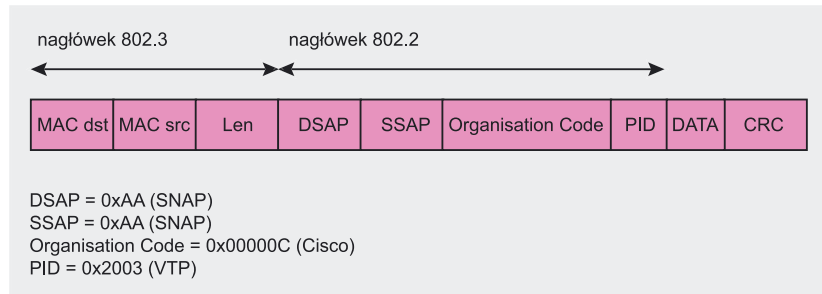
Obrona przed atakami IEEE 802.1Q jest taka sama, jak w przypadku DTP. Musimy wyłączyć automatyczne tworzenie magistral.

VTP (VLAN Trunking Protocol)

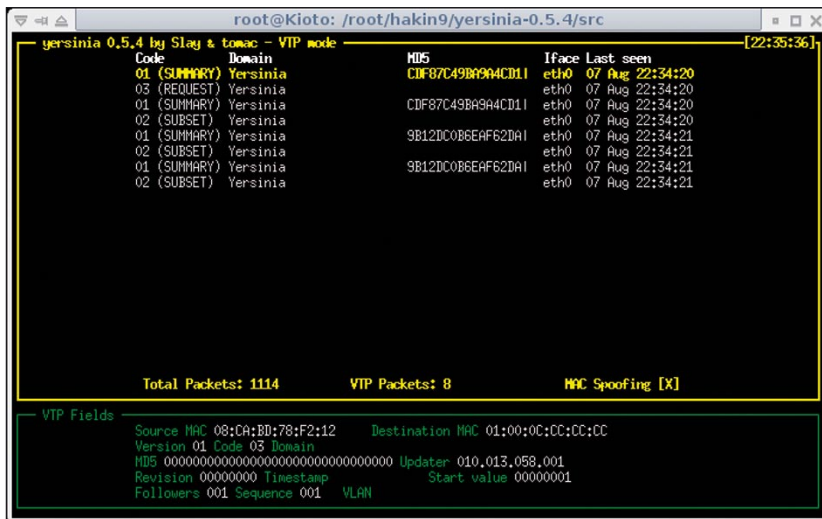
VTP to kolejny zastrzeżony protokół Cisco, przeznaczony do scentralizowanego zarządzania sieciami VLAN. Przykładowo jeśli VLAN jest skonfigurowana na przełączniku, związane z nią informacje (nazwa i identyfikator VLAN) mogą być skonfigurowane automatycznie we wszystkich przełącznikach należących do tej samej domeny VTP (patrz Ramka *Co to jest domena VTP*). Pakiet VTP są wysyłane z ramką IEEE 802.3 zawierającą nagłówek 802.2 SNAP na adres MAC *multicast* 01:00:0C:CC:CC:CC (patrz Rysunek 12).

Istnieją cztery typy pakietów VTP:

- *Summary advertisement* – **SUMMARY** – podobny do *Hello*, wysyłany co 5 minut,
- *Advertisement request* – **REQUEST** – używany przy żądaniu informacji,
- *Subset advertisement* – **SUBSET** – pakiet danych z opisami VLAN.
- *Join* – **JOIN**.



Rysunek 12. Struktura pakietu VTP



Rysunek 13. VTP attack results

Listing 12. Usunięcie VLAN – udany atak

```

zipi# sh vlan
VLAN Name                Status    Ports
-----
1    default                active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                         Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                         Fa0/9, Fa0/14, Fa0/15, Fa0/16
                                         Fa0/17, Fa0/18, Fa0/19, Fa0/24
                                         Gi0/1, Gi0/2
100  Office                 active    Fa0/10, Fa0/11, Fa0/12, Fa0/13
    
```

Ataki

Choć VTP umożliwia korzystanie z haseł, ta opcja nie jest domyślnie włączona. Jednak hash MD5 zawsze będzie obecny w pakiecie **SUMMARY** – jest obliczany razem z konfiguracją VLAN, hasłem (jeśli jest używane) oraz innymi polami.

Domyślnie pakiet VTP **SUMMARY** jest wysyłany co 5 minut. Biorąc pod uwagę fakt, że ataki VTP potrzebują nieco danych uzyskanych z pakietu tego typu, są nieco bardziej czasochłonne. Jednorazowy atak nigdy nie trwa jednak dłużej niż 5 minut.

Yersinia umożliwia kilka ataków VTP. Przyjrzyjmy się dwóm najważniejszym: dodawanie VLAN i usuwanie VLAN. Aby te ataki się powiodły, należy wykonać następujące kroki:



- zebranie przydatnych danych VTP; najlepszym sposobem jest oczekiwanie na pakiet *SUMMARY*,
- wysłanie pakietu *REQUEST*, by zebrać informacje o VLAN,
- modyfikacja sieci VLAN poznanych we wcześniejszym kroku,
- wysłanie nowej konfiguracji VLAN za pomocą pakietów *SUMMARY* i *SUBSET*.

Do dzieła

Konfiguracja przykładowego ataku jest taka sama, jak poprzednio – dwie sieci VLAN, *Office* (ID 100) i *Internet* (ID 200). Głównym celem ataku jest usunięcie tej drugiej sieci. Yersinia jest zlokalizowana na porcie Fa0/13.

Przejdźmy do ekranu VTP, wypełnijmy pola wartościami domyślnymi i rozpocznijmy atak *deleting one vlan*. Yersinia poprosi o podanie identyfikatora VLAN, który chcemy usunąć (w naszym przypadku 200). Po kilku minutach główne okno będzie wyglądać tak, jak na Rysunku 13. Jeśli atak się powiodł, nie zobaczymy VLAN 200 w konsoli switcha (patrz Listing 12).

Konsekwencje usunięcia VLAN mogą być różne. Testy wykazują, że wszystkie urządzenia połączone z portem należącym do usuniętej sieci VLAN zostają rozłączone. Zachęcamy jednak czytelników do własnych eksperymentów. Można spróbować skonfigurować jedną z maszyn tak, by pingowała komputer w tej samej sieci VLAN (na przykład bramę domyślną), a następnie usunąć VLAN. Zauważymy, że pakiety ICMP przestaną generować odpowiedzi. Jeśli jednak sieć VLAN zostanie ponownie dodana (oczywiście za pomocą Yersinii), *ping* znów zadziała.

Obrona

Tak, jak w przypadku dwóch poprzednich sekcji, wyłączenie automatycznego zestawiania magistral jest najbardziej skuteczną obroną. W tym wypadku możemy jednak również dobrze użyć haseł VTP.

Podsumowanie

Poznaliśmy całkiem sporo protokołów warstwy łącza danych, jak rów-

O autorach

David Barroso specjalizuje się w reagowaniu na incydenty w bezpieczeństwie sieciowym. Obecnie pracuje w hiszpańskiej firmie S21sec zajmującej się bezpieczeństwem sieci, jest głęboko zaangażowany w ogólnoswiatową społeczność zainteresowaną tym zagadnieniem – pisze artykuły, dokumenty i tworzy nowe narzędzia bezpieczeństwa.

Alfredo Andrés od kilku lat zajmuje się zawodowo bezpieczeństwem sieciowym. Jest aktywnym uczestnikiem ruchu Wolnego Oprogramowania, tworzy narzędzia i łąty. Pracuje w S21sec, szefując grupie zajmującej się testami penetracyjnymi.

Obaj autorzy prezentowali Yersinię na konferencji Black Hat Europe 2005, gdzie pokazali też, związany z jednym z omawianych protokołów, premierowy (0-day), kontrolowany atak na Cisco, odkryty podczas tworzenia narzędzia.

niez najpopularniejszych ataków na niego. Przykłady pokazują jasno, że warstwa druga może stanowić poważny problem, jeśli przy konfiguracji sieci nie weźmiemy pod uwagę

zagrożeń z nią związanych. Większość protokołów jest podatna na atak – podczas zarządzania nimi trzeba zachować szczególną ostrożność. ●

W Sieci

- <http://yersinia.sourceforge.net/> – strona domowa narzędzia Yersinia,
- <http://www.blackhat.com/presentations/bh-usa-02/bh-us-02-convery-switches.pdf> – prezentacja Seana Convery'ego o hakowaniu wartwy drugiej.

Spanning Tree Protocol

- <http://www.javvin.com/protocolSTP.html> – o *Spanning Tree Protocol*,
- <http://www.cisco.com/univercd/cc/td/doc/product/lan/trsrbf/frames.htm#xtocid61> – format pakietów BPDU,
- <http://www.cisco.com/warp/public/473/146.html> – artykuł *Understanding Rapid Spanning Tree Protocol (802.1w)*,
- <http://www.cisco.com/warp/public/473/17.html> – artykuł *Understanding Spanning-Tree Protocol Topology Changes*,
- <http://www.cisco.com/warp/public/473/65.html> – artykuł o *Spanning Tree Portfast BPDU Guard Enhancement*,
- <http://www.cisco.com/warp/public/473/74.html> – artykuł o *Spanning Tree Protocol Root Guard Enhancement*.

Cisco Discovery Protocol

- <http://www.cisco.com/univercd/cc/td/doc/product/lan/trsrbf/frames.htm#xtocid12> – format pakietów CDP,
- <http://www.phenoelit.de/stuff/CiscoCDP.txt> – rady grupy Phenoelit dotyczące podatności systemu IOS na ataki CDP.

Dynamic Trunking Protocol

- <http://www.netcraftsmen.net/welcher/papers/switchvtp.html> – artykuł *Switching: Trunks and Dynamic Trunking Protocol (DTP)*.

IEEE 802.1Q

- <http://standards.ieee.org/getieee802/download/802.1Q-2003.pdf> – standard IEEE 802.1Q w PDF.

VLAN Trunking Protocol

- <http://www.cisco.com/univercd/cc/td/doc/product/lan/trsrbf/frames.htm#xtocid31> – format ramki VTP.