

Struktura i historia protokołu IPSec

<http://ipsec.pl/ipsec/struktura-historia-protokolu-ipsec.html>

{W ciągu ponad 20 lat od swojego powstania protokół IP zrobił niemałą karierę. Zdecydowała o tym przede wszystkim jego elastyczność, pozwalająca równie skutecznie przenosić dane przez łącza o przepustowości 9600 bps, jak i kilkusetkrotnie szybsze. Łatwość adaptacji, połączona z takimi cechami jak możliwość korzystania z różnych tras w zależności od zmieniającego się stanu sieci, z pewnością zdecydowała o dzisiejszej popularności protokołu IP i opartych na nim protokołach wyższych warstw, stosowanych we współczesnym Internecie.

{Można śmiało powiedzieć, że protokół IP jest jednym z najlepiej zaprojektowanych protokołów w historii informatyki. Mało który z nich spełniał - i spełnia nadal - swoją funkcję nieprzerwanie przez tyle lat, podczas gdy oblicze Internetu zmienia się wokół z niespotykaną szybkością. Nie jest on jednak pozbawiony wad, które w dużym stopniu można jednak usprawiedliwić założeniami projektowymi i przewidywaniami co do rozmiaru sieci, jakie w czasie jego tworzenia były jak najbardziej uzasadnione.

{Pierwsze potencjalne ograniczenia IP zaczęto zauważać pod koniec lat 80-tych, kiedy niezwykle dynamiczny rozwój Internetu spowodował rozszerzenie jego granic poza sieci naukowe, komercjalizację i upowszechnienie w społeczeństwie. Lawinowy rozwój sieci wzbudził obawy po pierwsze co do zasobności globalnej puli adresowej, a po drugie problemów związanych z bezpieczeństwem sieci, która przestała być tylko miejscem przyjaznej współpracy naukowej.

{Zaczęto zdawać sobie sprawę że protokół IP, zaprojektowany podobno tak, by przetrwać wojnę jądrową nie chroni niestety przed atakami pochodzącymi z wewnątrz sieci. Zwrócono uwagę na możliwość podsłuchiwania ({sniffing}) i modyfikowania sieciowych transmisji na różne sposoby ({spoofing, {hijacking}). Wiele z tych problemów obchodzono na różnych innych poziomach, na przykład stosując oprogramowanie szyfrujące dane w warstwie aplikacyjnej (popularny program PGP, protokoły SSH i SSL), usprawnienia w systemach operacyjnych (nieprzewidywalne numery sekwencyjne TCP jako obrona przed {blind spoofingiem}) itp. Wkrótce stało się jednak jasne, że konieczne jest stworzenie mechanizmu, który zachowując elastyczność i inne zalety IP zapewniłby skuteczną ochronę nie tylko przed atakami aktywnymi (bezpośrednia ingerencja) i pasywnymi (bierny podsłuch), ale także na przykład przed analizą relacji przesyłanych danych ({traffic analysis}), mogąca dostarczać równie wartościowych wskazówek, nawet jeśli poddane jej informacje były zaszyfrowane.

{Wkroczenie biznesu do sieci i uczynienie jej równie popularnym środkiem komunikacji jak faks i telefon zaowocowało wreszcie koniecznością stworzenia mechanizmu, który pozwalałby odległym ośrodkom łączyć się za pomocą publicznego Internetu w sposób bezpieczny i wydajny. Konieczność ochrony całego ruchu wymianianego między oddziałami oraz stosowanie jednolitej adresacji, niezależnej od publicznie routowanych adresów IP, będącej podstawą VPN ({Virtual Private Networks}) były kolejnymi powodami, dla których ochronę należało przenieść do warstwy sieci. {h1}Początki bezpieczeństwa IPi/h1

{Doświadczenia uzyskane podczas kilkunastu lat wdrażania protokołu IP i wynikające z nich przewidywania co do przyszłych potrzeb pozwoliły IETF ({Internet Engineering Task Force}) na określenie założeń nowego protokołu {IP Security (IPSec)}, mającego uzupełnić funkcjonalność IP o ochronę danych. Nieprzypadkowo projekt ten zbiegł się z pracami grupy roboczej tworzącej kolejną wersję protokołu IP, nazywaną wówczas IPng ({IP Next Generation}). Nowe wcielenie IP, obecnie określane jako IPv6, poza szeregiem dalekowzrocznych usprawnień, takich jak rozszerzenie przestrzeni adresowej do 128 bitów (adresy IPv4 mają 32 bity) traktuje bowiem IPSec jako integralną część protokołu. Oznacza to, że każdy system operacyjny i urządzenie sieciowe (np. router) zgodne z IPv6 będzie także obsługiwać IPSec. Jest tutaj jednak pewien kruczek, o którym napiszemy dalej.

{Opracowaniem protokołu IPSec zajęła się w 1992 roku samodzielna grupa robocza IETF. Samodzielna po pierwsze dlatego, że zakres prac związanych ze stworzeniem protokołu o funkcjonalności IPSec był problemem samym w sobie, zaś ostateczne wpasowanie go w struktury IPv6 było już stosunkowo

proste. Kolejnym powodem było to, że użyteczna specyfikacja IPsec była potrzebna znacznie szybciej niż przewidywane wdrożenie IPv6, czyli pierwsze lata XXI wieku. Przy tym różnice w filozofii oraz budowie IPv4 i IPv6 były celowo na tyle niewielkie, by kluczowa część protokołu IPsec można było zastosować w obu wersjach IP.

{Dwoma podstawowymi zadaniami IPsec jest zapewnienie integralności oraz poufności danych przesyłanych przy pomocy IP, niezależnie od protokołów stosowanych w wyższych warstwach modelu ISO/OSI. Kolejny cel, zagwarantowanie autentyczności łączących się stron, jest realizowany do pewnego stopnia przez sam protokół IPsec i może być rozszerzany przez dodatkowe mechanizmy.

{Przez zapewnienie integralności rozumiemy tutaj możliwość wykrycia przypadkowych lub celowych modyfikacji wprowadzonych do przesyłanych informacji. Gwarancja ta jest realizowana przez zastosowanie kryptograficznych funkcji skrótu, co wyraźnie wyróżnia ten mechanizm od prostych sum kontrolnych protokołu IP i TCP. Te ostatnie pozwalają co prawda wykryć uszkodzenie pakietu powstałe wskutek niedoskonałości transmisji, ale nie zabezpieczają przed celowymi zmianami wprowadzonymi w złych intencjach.

{Poufność, czyli brak możliwości odczytania przechwyconych przez napastnika danych bez znajomości odpowiedniego klucza, jest zapewniana również przez algorytmy kryptograficzne w postaci szyfrów blokowych takich jak np. DES. Celowo nie wymieniamy w tym miejscu pozostałych szyfrów, ponieważ ich wybór jest w wysokim stopniu konfigurowalny, co będzie szczegółowo omawiane dalej.

{Kolejny postulat, czyli przezroczystość IPsec dla protokołów wyższych warstw (i na odwrót) jest zapewniana dzięki właściwości charakterystycznej dla modelu ISO/OSI, czyli enkapsulacji pakietów poszczególnych warstw w sobie (rysunek [1.1](#)). W praktyce rozszyfrowaniem i weryfikacją integralności otrzymanych z sieci danych zajmuje się system operacyjny, a dane trafiają do aplikacji, która nie musi nic wiedzieć o protokole IPsec. Jest to kolejna istotna cecha, odróżniająca ten protokół od innych metod stosowanych w wyższych warstwach.

{ >

{Z modelu tego wynika również istotna konsekwencja - podstawowa jednostka na której kończy i zaczyna się bezpieczny kanał zapewniany przez IPsec jest pojedynczy węzeł sieci ({host}), a nie na przykład aplikacja określonego użytkownika (jak w PGP czy S/MIME). W dobie rosnącego znaczenia komputerów osobistych coraz częściej bywa tak, że jeden węzeł to jeden użytkownik. Jak zobaczymy dalej, popularnym rozwiązaniem jest również szyfrowanie połączeń dopiero od pewnego miejsca, na przykład na wyjściu z sieci lokalnej.

{Pierwsze wersje specyfikacji IPsec zostały przedstawione w 1995 roku i zawierały opis dwóch oddzielnych podprotokołów składowych AH oraz ESP. Pierwszy z nich zapewniał integralność danych oraz do pewnego stopnia uwierzytelnienie stron transmisji. Drugi, ESP, zapewniał w początkowej wersji wyłącznie poufność, dopiero w późniejszym okresie dodano do niego dodatkowo ochronę integralności, podobną do tej zapewnianej przez AH. W tej postaci IPsec ujrzał światło dzienne w postaci dokumentów Request For Comments (RFC 1825 i następne), wyznaczających formalne standardy Internetu.

{W kolejnej oficjalnej wersji specyfikacji opublikowanej w 1998 roku (RFC 2401 i następne) zmieniono wiele drobnych szczegółów AH i ESP, co było odpowiedzią na komentarze otrzymane w międzyczasie. Równocześnie przedstawiono specyfikację rozbudowanego protokołu Internet Key Exchange, pozwalającego na automatyczne uzgadnianie parametrów kanałów IPsec pomiędzy węzłami, uwierzytelnienie węzłów i szereg innych zaawansowanych funkcji.

{Jak widać narodziny IPsec były długie i skomplikowane. W dużej mierze wynikało to z trudności merytorycznych, zrozumiałych przy konstruowaniu tak złożonego protokołu. Znacznym utrudnieniem były także ograniczenia natury polityczno-prawnej, związane ze stosowaniem kryptografii, które w ostatnim czasie stopniowo, choć powoli, zanikają. Między innymi amerykańskie ograniczenia na eksport oprogramowania szyfrującego były przyczyną stworzenia dwóch dublujących się częściowo algorytmów AH i ESP, co dodatkowo komplikuje architekturę IPsec. Pierwszy z nich, wykorzystujący wyłącznie kryptograficzne funkcje skrótu, z reguły traktowane bardziej liberalnie, miał

zapewniać ograniczone bezpieczeństwo tam, gdzie szyfrowanie danych było niemożliwe z powodu lokalnych zakazów lub braku możliwości wyeksportowania w pełni funkcjonalnych urządzeń.

{Stosowanie wyłącznie ochrony integralności ma jednak w niektórych wypadkach uzasadnienie praktyczne - przykładem może być tutaj komunikacja pomiędzy hostami, a serwerami DNS. Informacje udostępniane przez te ostatnie są z reguły publiczne i nie ma potrzeby ich szyfrowania, ważne jest natomiast by nie zostały one po drodze sfalszowane. Zabezpieczenie przed tego typu atakami (choć nie wszystkimi) zapewnia właśnie protokół AH lub, z wyłączonym szyfrowaniem, ESP. Jak widać, w przypadku IPsec niemal identyczny efekt można osiągnąć na więcej niż jeden sposób, co w przypadku protokołów bezpieczeństwa niekoniecznie musi być zaletą. Złożoność utrudnia stworzenie przejrzystej implementacji i cecha ta była już przedmiotem krytyki pod adresem IPsec.

{Warto przy okazji dodać, że to właśnie DNS miał być podstawa do stworzenia nowego, opartego o IPsec modelu komunikacji w Internecie, który planowano wdrożyć przy okazji prac nad bezpieczeństwem protokołu IP. DNS bowiem, nie będący niczym innym jak ogólnosiątkowa, rozproszona baza danych, mógł poza informacjami o adresach IP, serwerach pocztowych i tym podobnych udostępniać także klucze publiczne węzłów sieci. Host, chcący nawiązać połączenie z innym hostem mógłby poszukać w DNSie informacji o udostępnianych przez niego usługach związanych z bezpieczeństwem połączenia i na tej podstawie stwierdzić jego tożsamość, a następnie stworzyć chroniony przez IPsec kanał łączności. Na przeszkodzie ponownie stanęły niestety przeszkody natury formalnej, związane z wdrażaniem technik kryptograficznych, uniemożliwiając uruchomienie „bezpiecznego Internetu” w takiej skali, by miał on praktyczne znaczenie.

{Mimo to IPsec praktycznie od początku swojego istnienia był stosowany do budowy wirtualnych sieci prywatnych VPN ({Virtual Private Network}), co świadczy o dużym popycie na oferujący takie usługi protokoły. Zanim jeszcze oficjalna specyfikacja pierwszej wersji IPsec została opublikowana, firmy biorące udział w pracach nad protokołem wypuszczały urządzenia implementujące podzbiory powstającego protokołu. Niekoniecznie spełniające postulat wzajemnej kompatybilności, były i są one nadal wykorzystywane z powodzeniem do budowy VPN. Dobrym przykładem są tutaj urządzenia BorderGuard stworzone przez firmę Network Security Systems (obecnie SkyLine). Wraz z rozwojem IPsec implementacje te rozszerzono o kolejne części oficjalnej specyfikacji i w tej chwili większość urządzeń obsługujących ten protokół jest ze sobą kompatybilna.

Architektura IPsec - AH i ESP

{Jak już powiedzieliśmy, w modelu ISO/OSI protokół IPsec jest blisko związany z warstwą sieci czyli protokołem IP, w stosunku do którego jednak jest podrzędny. W praktyce oznacza to tyle, że pakiety poruszające się w sieci to zawsze nagłówek IP, a zaraz po nim enkapsulowany odpowiedni protokół bezpieczeństwa, AH lub ESP. Dopiero w takiej kopercie enkapsulowane są protokoły wyższych warstw.

{Protokół AH ({Authentication Header}) zapewnia ochronę integralności zarówno enkapsulowanego protokołu wyższej warstwy, jak i części nagłówka IP znajdującego się pod spodem. Ochrona obejmowana są te pola nagłówka, które nie ulegają zmianie podczas wędrówki przez sieć (adresy, identyfikator). Do zapewnienia integralności oraz, w pewnym stopniu wiarygodności stron połączenia wykorzystywane są kryptograficzne funkcje skrótu takie jak MD-5, SHA-1 czy RIPEMD-160 w trybie HMAC. Ten ostatni to wynik obliczenia skrótu przesyłanych danych oraz skonfigurowanego hasła, znanego tylko stronom połączenia.

{Protokół ESP ({Encapsulation Security Payload}) jest bardziej złożony, zapewnia bowiem szyfrowanie i ochronę integralności danych. Ta ostatnia jednak obejmuje wyłącznie dane wyższej warstwy, bez nagłówka IP. Szyfrowanie oraz uwierzytelnienie są opcjonalne, to znaczy strony mogą uzgodnić umowne algorytmy NULL w miejsce prawdziwych szyfrów lub funkcji skrótu. Algorytmy funkcji skrótu są dokładnie takie same jak w AH (i też w trybie HMAC), natomiast poufność zapewniają szyfry blokowe w trybie CBC, takie jak DES, 3DES, Blowfish, CAST-128 i - od niedawna - Rijndael/AES.

Tryby stosowania IPsec

{Jeśli przyjrzymy się bliżej możliwym do znalezienia w sieci pakietom IPsec, to stwierdzimy że występują one zawsze w dwóch typach, narysowanych poniżej. Różnią się one kolejnością enkapsulowanych nagłówków i, jak się za chwilę okaże, zakresem oferowanej ochrony.

{

{Na rysunku [2a.](#) widzimy najbardziej intuicyjną wersję zastosowania protokołu IPsec, czyli tzw. tryb transportowy (transport mode). Pomiedzy nagłówkiem IP a protokołem wyższej warstwy dodany został nagłówek IPsec, chroniący enkapsulowane wewnątrz dane. Tak skonstruowany pakiet może poruszać się po globalnej sieci, ale mimo to tryb transportowy stosuje się prawie wyłącznie w sieciach lokalnych. Powodem są wymagania protokołu IPsec jeśli chodzi o kolejność pakietów docierających do docelowego routera. Wymóg ten może być zniweczony przez możliwość w heterogenicznej sieci fragmentacji oraz różne trasy, którymi mogą wedrować poszczególne pakiety. W sieciach lokalnych, gdzie problem ten nie występuje, tryb transportowy doskonale spełnia swoją rolę i część implementacji, szczególnie w kartach sieciowych, obsługuje tylko ten wariant.

{Tryb tunelowy (tunnel mode), uwidoczniony na rysunku [2b.](#) jest połączeniem enkapsulacji IP w IP z opakowaniem przenoszonych pakietów w IPsec. W ESP lub AH enkapsulowany jest kompletny pakiet IP, czyli pakiet wyższej warstwy wraz z nagłówkiem IP. Adresy źródłowy i docelowy umieszczone w nagłówku IP znajdującym się pod IPsec są z reguły adresami odpowiednich routerów szyfrujących (security gateways), łączących ze sobą tworzące VPN sieci lokalne znajdujące się w odległych lokalizacjach. Zauważmy, że dzięki temu że wewnętrzny pakiet jest enkapsulowany w całości, bierny obserwator nie ma nawet możliwości stwierdzenia między jakimi adresami IP w VPN zachodzi komunikacja. [Szyfrowane tunele SA/h1](#)

{Termin ten, którego najbliższym polskim odpowiednikiem jest chyba „bezpieczny tunel”, jest jednym z podstawowych pojęć leżących u podstaw architektury IPsec. W istocie jest to jednokierunkowy kanał, którego końcami są dwa hosty, identyfikowany przez unikalny numer SPI (Security Parameters Index).

{Numer SPI przekłada się na cały zestaw parametrów, charakteryzujących dany kanał (algorytm szyfrujący, algorytm uwierzytelnienia, „okno” chroniące przed powtarzaniem pakietów, okres ważności i in.) znanych wyłącznie hostom będącym końcami tunelu. SPI jest w praktyce dowolna, 32-bitowa liczba, ustalana arbitralnie przez administratora podczas ręcznej konfiguracji tunelu, lub wybierana losowo w razie konfiguracji automatycznej. Identyfikator ten jest jedynym charakterystycznym parametrem tunelu, który jest widziany przez osobę potencjalnie podsłuchującą łącza. Co więcej, informacja ta ma konkretne znaczenie (algorytm szyfrujący itp.) wyłącznie dla routerów, stanowiących końce tunelu.

{Dlaczego kanały SA są jednokierunkowe? Takie podejście upraszcza ich konstrukcję i powoduje, że ich konfiguracja jest bardziej elastyczna. Zwykle pełnowartościowy tunel IPsec będzie składał się z dwóch SA, po jednym w każdym kierunku, jednak nic nie stoi na przeszkodzie by tworzyć tunele szyfrowane tylko w jedną stronę lub inne kombinacje kanałów. Istnienie dwóch kanałów powoduje również, że ruch w każdym kierunku jest (a raczej może być) szyfrowany innym kluczem i może mieć inny okres ważności, co umożliwia precyzyjne dopasowania parametrów tunelu do charakterystyki ruchu.

{Co więcej, kanały SA mogą się wzajemnie w sobie zawierać i nie muszą się zaczynać w tych samych miejscach (na tych samych hostach). Możliwe jest np. enkapsulowanie ESP w AH, choćby w taki sposób jak przedstawiono na rysunkach [3b.](#) i [3c.](#)

{``

{Jak wygląda konfiguracja SA w praktyce? Przeważnie polega to na dodaniu do systemowej bazy SAD (SA Database) pozycji o mniej więcej takiej wymowie: „{kanał [1234](#) zaczyna się a hoście [A](#), kończy na [B](#) i jest szyfrowany algorytmem [E](#) z kluczem [XYZ](#)”. Fakt istnienia takiego SA w systemie nie oznacza jeszcze, że dane w tej relacji (z hosta A do B) zostaną w istocie zaszyfrowane. Decyzja o tym zostaje podjęta na podstawie innej systemowej bazy, SPD (Security Policy Database). [IPsec a polityka bezpieczeństwa/h1](#)

{SPD jest przełożeniem administracyjnie narzuconej polityki bezpieczeństwa danego systemu na implementację IPsec i ma wobec niej rolę nadrzędną. W SPD znajdują się informacje, które mogą być przepisane niemal dosłownie z „papierowej” polityki bezpieczeństwa, na przykład: „{cały ruch wychodzący do sieci N [musi](#) być szyfrowany”. SPD pozwala na gradację

wymogów, a więc zamiast „musi” można zastąpić przez „może” (w domyśle: „jeśli uda się stworzyć odpowiedni kanał”). Można określić kierunek, w odniesieniu do ruchu przychodzącego i wychodzącego. Można wreszcie sprecyzować wymogi według adresów hostów, sieci, numerów portów i rodzajów protokołów.

{Na czym polega nadrzędna rola SPD? W poprawnej implementacji IPSec pakiet, który nie spełnia wymogów SPD nie ma prawa opuścić danego systemu. W najbardziej restrykcyjnym przypadku oznacza to, że łączność będzie nawiązana albo bezpiecznym kanałem, albo w ogóle. }h1;Przetwarzanie pakietów IPSec;/h1;

{Niejako mimochodem poznaliśmy dwie najważniejsze systemowe bazy danych wykorzystywane przez IPSec - SAD i SPD. Jak są one powiązane ze sobą i jak wygląda w praktyce obsługa pakietu przez stos IP/IPSec?

{Rozważmy następujący scenariusz: do stosu IP trafia przeznaczony do wysłania pakiet, zawierający jakieś dane od hosta 10.0.0.8 dla hosta 10.1.1.123. System najpierw poszukuje w bazie SPD pozycji najlepiej pasującej do otrzymanego pakietu. Znajduje tam wpis mówiący, że wymagane jest szyfrowanie protokołem ESP w trybie transportowym wszystkich pakietów o adresie docelowym w sieci 10.1.1.0/24.

{Następny krok, to sprawdzenie w bazie SAD, czy istnieje jakiś kanał SA dla tej relacji. Tym razem dopasowanie musi być dokładne, to znaczy musi istnieć kanał przeznaczony konkretnie dla pakietów od hosta 10.0.0.8 do 10.1.1.123. Musi on mieć unikalny numer SPI, skonfigurowany algorytm szyfrujący, klucz i inne parametry.

{Jeśli taki kanał istnieje, to pakiet jest nim po prostu wysyłany i dane w bazie SAD są uaktualniane (licznik bajtów, pakietów, wektor inicjalizujący IV itp). Pomińmy chwilowo przypadek, kiedy w SAD nie ma odpowiedniego kanału i zobaczmy, jak przetwarzane są pakiety przychodzące.

{System otrzymuje pakiet zapakowany w protokół ESP, którego kluczowym oznaczeniem jest, jak pamiętamy, numer SPI. Teraz ponownie przeszukiwana jest baza SAD, tym razem w poszukiwaniu kanału legitymującego się SPI takim, jak otrzymany pakiet. Jeśli takie SA istnieje, to pakiet jest rozszyfrowywany za pomocą znajdujących się tam informacji (klucza, IV). Na tym etapie system wykryje błąd szyfrowania lub uwierzytelnienia pakietu i odrzuci go jako fałszywy. Co jednak stanie się, jeśli w SAD nie będzie kanału o pasującym SPI?

{Jeśli system nie udostępnia żadnych dodatkowych mechanizmów pozwalających na automatyczne tworzenie kanałów bezpieczeństwa, to taki pakiet zostanie po prostu odrzucony, wraz z odpowiednim komunikatem w logach systemowych. Z taka sytuacją mamy do czynienia w przypadku, gdy administrator ręcznie konfiguruje na każdym z węzłów sieci polityki bezpieczeństwa oraz odpowiednie kanały do innych hostów. Pojawienie się w sieci pakietu bez przypisanego kanału bezpieczeństwa jest zdarzeniem oznaczającym albo błąd konfiguracji, albo próbę ataku i w każdym wypadku jest godne uwagi.

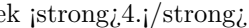
{Zauważmy przy tym, że wariant z ręczną konfiguracją może mieć sens jedynie dla sieci złożonej z niewielu węzłów. Biorąc pod uwagę, że każde faktyczne połączenie składa się w praktyce z dwóch lub więcej kanałów skonfigurowanie sieci do której podłączonych ma być kilkanaście węzłów korzystających z IPSec jest skomplikowane. Jeśli dodatkowo komunikacja ma się odbywać w modelu {mesh (każdy z każdym), to liczba kombinacji kanałów SA gwałtownie wzrasta, bardzo utrudniając jakiegokolwiek zmiany w przyszłości, na przykład okresową wymianę kluczy kryptograficznych. }h1;Automatyczna negocjacja parametrów bezpieczeństwa;/h1;

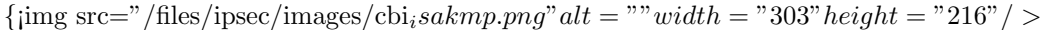
{Aby temu zaradzić grupa robocza IPSec równolegle pracowała nad propozycją protokołu automatycznej negocjacji parametrów bezpieczeństwa, możliwego do wykorzystania z IPSec. Propozycji takich pojawiło się przynajmniej kilka, wśród których wymienić warto protokoły SKIP firmy Sun, Photuris oraz IKE ({Internet Key Exchange).

{Głównym zadaniem wszystkich tych protokołów jest wzajemne uwierzytelnianie hostów nawiązujących połączenia po IPSec, a następnie uzgadnianie krótkoterminowych kluczy kryptograficznych na potrzeby poszczególnych kanałów SA. Jedną i drugą funkcją jest realizowana na podstawie skonfigurowanych na stałe danych uwierzytelniających. Te ostatnie mogą być różne, w zależności od

protokołu - na przykład hasła wspólne między parami hostów ({shared secret}), certyfikaty X.509 czy klucze PGP. SKIP i Photuris umożliwiają wyłącznie uwierzytelnienie na podstawie hasła. Protokół IKE obsługuje natomiast wszystkie wyżej wymienione metody i zostawia jeszcze miejsce na prywatne rozszerzenia. Jest on używany najszerzej i jemu przyjrzymy się trochę bliżej, jednak szczegółowe omówienie jest tematem na osobny artykuł.


{IKE składa się z dwóch części: ISAKMP ({Internet Security Association and Key Management Protocol}), stanowiącego faktyczny protokół negocjacji parametrów IPsec oraz Oakley, będącego kryptograficznym protokołem wymiany kluczy za pomocą algorytmu Diffie-Hellmana. ISAKMP stanowi faktyczny trzon całości i z tego powodu nazwy tej używa się niekiedy zamiennie z IKE.

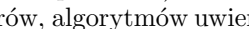
{Wymiana za pomocą IKE następuje dwuetapowo. Najpierw ustalana jest tożsamość komunikujących się węzłów i tworzony jest bezpieczny kanał (tzw. ISAKMP SA), utrzymywany przez cały czas trwania sesji i służący następnie do faktycznej negocjacji parametrów IPsec, które mogą być później zmieniane za pomocą tego samego kanału. Poza wymianą kluczy węzły mogą uzgodnić listę obsługiwanych przez siebie algorytmów szyfrujących, co w dużym stopniu ułatwia budowanie heterogenicznych sieci. Uproszczony schemat kolejnych etapów nawiązywania komunikacji przy pomocy ISAKMP przedstawia rysunek 

{ src="/files/ipsec/images/cbi_sakmp.png" alt = "" width = "303" height = "216" / >

{Jak już pisaliśmy, uwierzytelnienie w IKE może być dokonywane na różne sposoby. W najprostszym przypadku każda para węzłów musi mieć ustalone wspólne hasło, które służy następnie do obliczania kluczy metoda Diffie-Hellmana. Konieczność konfigurowania hasła na wszystkich węzłach jest tutaj pewnym ograniczeniem, ponieważ wymaga wiedzy, jakie hosty będą się między sobą komunikować. W przypadku dużych sieci konfiguracja taka może być również dość pracochłonna.

{Zastosowanie certyfikatów X.509, czyli kluczy publicznych podpisanych przez nadrzędny urząd certyfikujący ({Certifying Authority), nie ma tych ograniczeń i bardzo ułatwia budowanie dużych oraz zróżnicowanych sieci. Daje także dostęp do ogromnych możliwości PKI ({Public Key Infrastructure), stającego się powoli światowym standardem. W najbardziej liberalnym przypadku dany węzeł nie musi wiedzieć nic o innych węzłach z którymi będzie się łączył, lub które będą się łączyć z nim. Po rozpoczęciu komunikacji, ale przed uzgodnieniem ISAKMP SA może on zweryfikować autentyczność certyfikatu partnera dzięki podpisowi CA. Wymaga to oczywiście zainstalowania na węzle klucza publicznego urzędu, ale będzie to jeden i ten sam klucz na wszystkich węzłach.

{Korzyści płynące ze stosowania protokołów automatycznej negocjacji IPsec są zauważalne szczególnie w przypadku dużych sieci, pozwalają także łatwo osiągnąć efekty, które w praktyce byłyby niemożliwe do wykonania w sieciach konfigurowanych ręcznie. Jedną z takich funkcji jest automatyczna renegotiacja kluczy kryptograficznych co określony czas. Operacja ta trwa stosunkowo krótko (może być więc wykonywana często) i gwarantuje, że w razie naruszenia bezpieczeństwa systemu, ujawnione mogą zostać jedynie dane przechwycone  włamaniu. Cecha ta, określana jako {Perfect Forward Security chroni przed sytuacją gdy atakujący zapisuje wszystkie przechwycone w przeszłości dane w nadziei, że kiedyś uda mu się zdobyć klucz do ich rozszyfrowania. W przypadku renegotiacji kluczy (tworzonych na podstawie liczb losowych) przeszłe klucze zostają po prostu zapomniane i włamywacz nie znajdzie ich w systemie nawet w przypadku całkowitego opanowania danego węzła.

{Niemniej ważne są korzyści administracyjne, które umożliwiają na budowanie dużych sieci przy zachowaniu bezpieczeństwa zapewnianego przez IPsec. Wspomniana już funkcja negocjacji i wyboru stosowanych algorytmów sprzyja kompatybilności implementacji różnych producentów w heterogenicznych sieciach. Protokół ISAKMP jest także łatwo rozszerzalny i zmieniając zdefiniowane dla internetowej wersji DOI ({Domain of Interpretation) można go przystosować całkowicie do potrzeb własnej instytucji (własny zestaw szyfrów, algorytmów uwierzytelnienia). 

{Protokół IPsec nie zbliżył się niestety do swojego kuzyna, IP, jeśli chodzi o doskonałość projektu. Wprost przeciwnie, praktycznie od początku był krytykowany za rozmaite niekonsekwencje oraz, przede wszystkim, za nadmierne skomplikowanie. Niektóre błędy zostały usunięte w wersji IPsec opublikowanej w 1998 roku, na przykład część potencjalnych furtek do ataków typu DOS ({Denial of Service), częściowo na podstawie sugestii autorów innych protokołów (w tym Photurisa).

{Zarzut złożoności jest zapewne najpoważniejszy i trudno go usprawiedliwić zapewniana funkcjonalnością, bo często jest ona dublowana przez różne elementy protokołu. Na przykład ochrona integralności zapewniana jest niemal w równym stopniu przez ESP i AH (usunięcie tego ostatniego ze specyfikacji postulowano już kilkakrotnie).

{Również sama specyfikacja IPsec krytykowano za brak przejrzystości i niejednoznaczność niektórych sformułowań. Wszystko to powoduje, że IPsec nie jest protokołem uznawanym za dobrze zaprojektowany, choć jest on wykorzystywany powszechnie i praktycznie nie ma alternatywy. Te sytuacje dobrze charakteryzuje głos dwóch z krytyków IPsec, Nielsa Fergussona i Bruce Schneiera, którzy w 1999 roku opublikowali analizę protokołu: „Nawet pomimo dość poważnych zarzutów jakie wysuneliśmy wobec IPsec, jest on prawdopodobnie najlepszym protokołem bezpieczeństwa z obecnie dostępnych. W przeszłości przeprowadziliśmy podobne analizy innych protokołów o analogicznym przeznaczeniu (w tym PPTP). Żaden ze zbadanych protokołów nie spełnił swojego celu, ale IPsec zbliżył się do niego najbliższej. (...) Mamy ambiwalentne odczucia wobec IPsec. Z jednej strony IPsec jest znacznie lepszy niż jakikolwiek protokół bezpieczeństwa IP stworzony w ostatnich latach: Microsoft PPTP, L2TP itp. Z drugiej strony nie wydaje nam się, by zaowocował on kiedykolwiek stworzeniem w pełni bezpiecznego systemu.”

{Pesymizm badaczy wydaje się być przynajmniej po części uzasadniony, jednak wszystkie błędy protokołu nie mają raczej charakteru otwartych dziur, grożących złamaniem bezpieczeństwa sieci, ale są za to dość liczne i ułatwiają powstawanie potencjalnych słabości w samych implementacjach, jeśli twórcy nie zadbają o to na własną rękę. Trudno powiedzieć jak to wygląda w przypadku implementacji dostępnych obecnie na rynku, ponieważ nie zostały opublikowane wyniki audytu żadnej z nich, nie wiadomo także czy takie audyty są prowadzone. Również certyfikaty przyznawane przez instytucje takie jak <http://www.icsa.net/html/communities/ipsec/> ICSA, są raczej stwierdzeniem zgodności ze specyfikacją niż faktycznej poprawności implementacji, która w przypadku systemów kryptograficznych ma bardzo duże znaczenie.

{Protokół IPsec znajduje obecnie najszersze zastosowanie w wymagających poufności połączeniach B2B (Business to business) - pomiędzy oddziałami korporacji, pomiędzy partnerami handlowymi czy wreszcie pomiędzy firmą, a jej klientami (Business to customer). Z technicznego punktu widzenia są to wszystko rozległe sieci prywatnych (VPN), w których połączenia pomiędzy poszczególnymi LANami są chronione przez brzegowe routery szyfrujące. Istnieje również szereg urządzeń, umożliwiających zdalny dostęp do firmowej sieci pracownikom łączącym się z obcych routerów dostępowych. Jednym z najdłuższych obecnych na rynku urządzeń tego typu są wspomniane już routery i serwery dostępowe IPsec z serii SkyLine. Rozwiązanie tego typu oferują również firmy Intel, Cisco, Red Creek i inne. Z rodzimych propozycji można jedynie wymienić rodziny, jakim jest router szyfrujący Cryptonite krakowskiej firmy ABA. Jeśli chodzi o systemowe implementacje IPsec, to można je znaleźć we wszystkich nowoczesnych systemach operacyjnych - od dostępnych na zasadzie open-source systemów BSD i Linuxa, po komercyjne systemy takie jak Solaris 8.

{Wydaje się, że w chwili obecnej IPsec jest najbezpieczniejszym i najbardziej przenośnym sposobem na stworzenie bezpiecznej sieci korporacyjnej opartej o łącza publiczne. Wzrastające zainteresowanie tym protokołem firm oraz urzędów publicznych pozwala sądzić, że jego rola w bezpiecznej telekomunikacji będzie konsekwentnie wzrastać. Pomimo wspomnianych wyżej wad protokołu IPsec sprawia wrażenie najlepszej propozycji, pod względem skalowalności i bezpieczeństwa wyprzedzającym konkurencyjne techniki tego typu.

{Artykuł ukazał się po raz pierwszy w marcu 2001 w magazynie Bezpieczeństwo IT.