heise Security

Dieser Artikel erschien ursprünglich in c't 07/2006, Seite 114

VPN-Knigge

VPN-Protokolle und Standards

Daniel Bachfeld - *13.04.2006*

Verschiedene Techniken bieten sich zum Aufbau eines virtuellen privaten Netzwerks an, aber nicht alle passen zum gedachten Einsatzszenario. Eine Übersicht über Standards und Protokolle erleichtert die Auswahl.

Das derzeit am häufigsten eingesetzte VPN-Protokoll ist IPSec (IP Security). Mittlerweile findet man es so gut wie in jedem Firewall-Produkt, einige Heim-Router im oberen Preissegment haben es eingebaut und seit Windows 2000 ist es Bestandteil von Microsofts Betriebssystem. Mit IPSec ist es möglich, IP-Pakete kryptographisch gesichert über öffentliche Netze zu transportieren. Mehrere RFCs (RFC2401 - 2409) beschreiben die bei IPSec zum Einsatz kommenden Verfahren und Protokolle.

IPSec bietet zwei Sicherungsarten: Authentication Header (AH) und Encapsulation Security Payload (ESP) jeweils in Kombination mit den Betriebsmodi Tunnel oder Transport. Mit AH kann der Anwender nur die Integrität und Echtheit der Daten sicherstellen, indem über jedes verschickte Paket ein HMAC (siehe **Glossar [1]**) gebildet wird. AH wird nur selten eingesetzt, da es kaum Anwendungen gibt, in denen nur die Integrität zählt.

Für die Vertraulichkeit der Kommunikation sorgt ESP, das die Pakete verschlüsselt. Zusätzlich schützt eine Integritätssicherung vor Manipulationen, allerdings nicht für das gesamte Paket wie bei AH. Bei ESP fließt etwa die IP-Adresse nicht in die Berechnung des HMAC ein, sodass sich diese manipulieren lässt. Dies erlaubt aber trotzdem kein IP-Spoofing, da eine Authentisierung der Kommunikationspartner beim Tunnelaufbau stattfindet.

Beim Transport-Mode bleiben die äußeren Header erhalten.

Zusätzlich zur Wahl zwischen AH und ESP hat der Anwender die Möglichkeit, die Pakete im Transport- oder Tunnel-Mode über das Netz zu verschicken. Beim Transport-Mode wird der Original-IP-Header, also IP-Adresse plus IP-Optionen, weiter benutzt. Im Tunnel-Mode kapselt IPSec das ganze Paket samt IP-Header und schreibt einen neuen IP-Header davor -- die Original-IP-Adresse ist nicht mehr sichtbar. Erst bei der Entschlüsselung auf der gegenüberliegenden Seite kommt die IP-Adresse mitsamt des restlichen Paketes wieder zum Vorschein.

Üblicherweise kommt die Kombination ESP und Tunnelmode auf VPN-Gateways zum Einsatz, wenn entfernte Subnetze miteinander über ein unsicheres Netz gekoppelt werden. Sollen zwei Rechner miteinander über IPSec im LAN kommunizieren, so wählt man meist den Transport-Mode.

Der Tunnel-Mode verschleiert die Original-Adresse des Absenders.

Schlüsselweltwurf

Die kryptographischen Funktionen von AH und ESP beruhen auf symmetrischen Schlüsseln. Um diese nicht vorab austauschen zu müssen, handelt das Internet-Key-Exchange-Protokoll (IKE) diese beim Aufbau der Verbindung dynamisch aus. Nebenbei erledigt IKE auch noch die Authentifizierung der Teilnehmer und das Aushandeln der Security Associations (SA), in denen die Konfiguration der Verbindung festgehalten wird.

Beim Verbindungsaufbau durchläuft IKE zwei Phasen: In Phase 1 (Main Mode) tauschen die Partner in vier Nachrichten Schlüsselmaterial aus, um sich auf einen gemeinsamen symmetrischen Schlüssel (SKEYID) zu einigen. Aus SKEYID werden ein Schlüssel zur Authentisierung und einer zur Verschlüsselung der weiteren IKE-Nachrichten abgeleitet sowie ein Schlüssel für die spätere Phase 2. Anschließend erfolgt über zwei weitere, nun verschlüsselte Nachrichten die Authentifizierung der VPN-Teilnehmer durch digitale Signaturen, RSA-Schlüssel oder Pre-Shared Keys (PSK). Letztere sind nichts anderes als geheime Passwörter, die auf beiden Seiten der IPSec-Verbindung identisch sein müssen.

PSKs sind wesentlich einfacher zu handhaben als Zertifikate, haben aber im Main Mode einen entscheidenden Nachteil: Sie funktionieren nur mit statischen IP-Adressen. Das liegt daran, dass im Main Mode ein PSK fest an die IP-Adresse der Gegenstelle gekoppelt ist und nicht nur zur Authentifizierung dient. Der PSK geht nämlich in die Berechnung des Schlüssels SKEYID ein. Ohne die feste Zuordnung könnte eine Gegenstelle die beiden letzten chiffrierten Nachrichten nicht entschlüsseln. Im Unterschied dazu wird bei der Authentifizierung durch Zertifikate der Schlüssel mittels des Diffie-Hellman-Key-Exchange-Verfahrens immer neu berechnet.

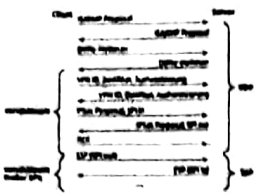
Dieser Artikel ist eine gekürzte Fassung des ursprünglich in c't 07/06, Seite 114, erschienen Artikels.

Schnell, schnell

Um PSKs mit dynamischen IP-Adressen nutzen zu können, kommt der auf drei Nachrichten verkürzte Aggressive Mode zum Einsatz. Dort sind die PSKs an so genannte Peer-IDs gebunden, beispielsweise eine Mail-Adresse oder einen Domain-Namen, die unverschlüsselt übertragen werden. Daran kann der VPN-Partner erkennen, welches der richtige PSK ist. Prinzipiell sieht der RFC2409 (multiple PSKs) vor, sodass beispielsweise jeder mobile Client mit einem individuellen Key ausgestattet werden kann. Leider unterstützt dies nicht jeder Hersteller, sodass dann alle Clients denselben PSK benutzen müssen. Wird ein Client kompromittiert, müssen alle anderen Clients den PSK wechseln.

Die im Aggressive Mode benutzte Klartextübertragung bietet einen Angriffspunkt: Zur Authentifizierung sendet das Gateway einen aus dem PSK abgeleiteten Hashwert über das Netz. Da dieser Hash nicht verschlüsselt ist, lässt sich damit unter Umständen der Schlüssel über Wörterbuch- oder Brute-Force-Angriffe rekonstruieren. Aus Sicherheitsgründen sollten man daher den Main Mode nutzen **[1 [2]]**.

Über die in Phase 1 aufgebaute gesicherte Verbindung können die VPN-Peers sich nun in Phase 2 (Quick Mode) auf einen symmetrischen Schlüssel für die IPsec-Verbindung einigen. Danach steht der IPSec-Tunnel, durch den etwa ein Gateway IP-Pakete routet, die für das LAN hinter dem gegenüberliegenden Gateway bestimmt sind.



Nach maximal neun Nachrichten ist ein IPSec-Tunnel aufgebaut. Im Aggressive Mode steht der Tunnel nach sechs Paketen.

Fallstricke

Da der IPSec-Standard bereits einige Jahre auf dem Buckel hat, sind dort einige Netzkonfigurationen nicht berücksichtigt. So hat IPSec immense Probleme mit dem heute in vielen Netzen eingesetzten NAT, weil dabei das IPSec-Paket verändert wird. Je nach NAT-Art (Basic NAT oder Network Adress Port Translation, NAPT) erhält ein Paket eine neue IP-Adresse und gegebenenfalls noch eine neue Quell-Portnummer. AH, egal ob im Transport- oder Tunnel-Mode, streckt hier sofort die Waffen. Weil der Paket-Header verändert wurde, stimmt der HMAC nicht mehr.

Bei ESP ist es etwas komplizierter: Um Ports umzuschreiben, müsste ein NAT-Router den TCP/UDP-Header lesen können. Der Original-Header ist aber verschlüsselt, sodass eine Zuordnung unmöglich ist. Mit ESP im Tunnelmode würde NAT zwar klappen, vorher scheitert aber schon IKE an NAT. Denn IKE kommuniziert fest über den UDP-Quell- und Zielport 500. Wird der verändert, kommt keine Verbindung zu Stande. Einige Router unterstützen deshalb das IPSec-Passthrough-Verfahren, bei dem die IKE-Ports nicht verändert werden. Zudem leitet der Router ESP-Pakete damit richtig weiter. Da die ESP-Pakete nur einer Verbindung zugeordnet werden können, funktioniert Passthrough nur mit einem einzigen Client. Um sich nicht auf den Router verlassen zu müssen, ist das ursprüngliche IPSec daher kaum noch gebräuchlich. Vielmehr setzt man es mit der IPSec-Erweiterung NAT-Traversal ein (RFCs 3947 und 3948). Dabei tauschen beide Seiten über das NAT-Traversal-Protokoll verschiedene Informationen aus. Anschließend werden ESP-Pakete in UDP-Pakete verpackt und über Port 4500 verschickt. Nun können NAT-Router ohne Probleme sowohl IP-Adressen als auch Ports umschreiben.

Ablösung

Weil der Aufbau von IPSec-VPNs relativ komplex und fehleranfällig ist, setzen sich einfachere Lösungen durch, die zur Sicherung auf die Standards Secure Socket Layer (SSL) beziehungsweise Transport Layer Security (TLS) setzen. Der Begriff SSL-VPN hat derzeit zwei Bedeutungen: Zum einen bezeichnet clientless SSL-VPN den Web-Zugriff von entfernten Anwendern per SSL-Verbindung auf einen Server, die bestimmte Applikationen anbietet, etwa Datenbankzugriffe mittels Webbrowser [2 [3]] . Der Vorteil: https ist in jedem Browser eingebaut, ist erprobt und kommt überall durch. Der Nachteil: Um auch mit nicht webbasierenden Anwendungen zu arbeiten, müssen auf Client- und/oder Serverseite Umsetzer mitlaufen, etwa Java-Plug-ins, die deren Daten über die Browserverbindung umleiten. Zum anderen sind damit Lösungen wie OpenVPN gemeint, die IP-Pakete transparent tunneln und somit völlig unabhängig von der Anwendung sind.

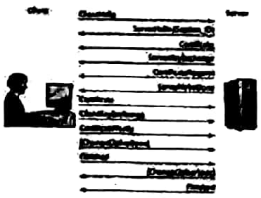
SSL und TLS unterscheiden sich kaum: Das ursprünglich von Netscape entwickelte SSL wurde ab Version 3.0 mit einigen kleineren Änderungen von der IETF übernommen und TLS 1.0 genannt (RFC 2246). TLS unterstützt zur Authentisierung der Daten HMAC und erzeugt das Schlüsselmaterial mit einer anderen Funktion als SSL (PRF statt RAND). Bei der Nachrichtenübermittlung gibt sich TLS als SSL-Version 3.1 zu erkennen.

Der TLS Record sorgt für die verschlüsselte Übertragung der Application Data.

Das TLS-Protokoll nutzt zwei Schichten: Den Record Layer und die darauf aufsetzenden Protokolle Alert, Change Cipher Spec, Handshake und Application Data. Über das TLS-Handshake-Protokoll einigen sich die Peers auf einen individuellen symmetrischen Sitzungsschlüssel, beispielsweise wie bei IPSec mit Hilfe von Diffie-Hellman, und mit welchen Algorithmen verschlüsselt und authentisiert werden soll. TLS nutzt insgesamt vier Schlüssel: je einen zum Ver- und Entschlüsseln sowie je einen zur Authentisierung ankommender und abgehender Pakete.

Zur Authentifizierung dienen in der Regel Zertifikate, TLS unterstützt aber auch unsignierte RSA-Keys. Im Internet trifft der Anwender meist auf unidirektionale Authentifizierung: Nur der Server beglaubigt mit dem SSL-Zertifikat seine Identität. Bei SSL-VPNs beweist hingegen auch der Client seine Identität mit einem Zertifikat.

Mit dem Change-Cipher-Spec-Protokoll signalisiert ein Teilnehmer, dass alle nun folgenden Pakete mit dem ausgehandelten Sicherheitskontext, beispielsweise AES-CBC mit 256 Bit und SHA1, zu schützen sind. TLS unterstützt fast 40 Sicherheitskontexte verschiedener Kombinationen von RSA, DH, AES, DES, SHA-1 und MD5. Geht an irgendeiner Stelle etwas schief, so informieren sich die Peers gegenseitig über das Alert-Protokoll.



Der TLS-Verbindungsaufbau wird mit maximal 13 Nachrichten in vier Paketen abgewickelt.

Der TLS-Verbindungsaufbau wird mit maximal 13 Nachrichten abgewickelt, die über zwei Server-Pakete und zwei Client-Pakete verschickt werden. Steht der Tunnel, so komprimiert und verschlüsselt der TLS Record alle Daten des Application Data Layer und reicht sie weiter an die TCP-Schicht.

Offen und geschlossen

Die Open-Source-Lösung OpenVPN nutzt ebenfalls TLS, geht aber ein paar Umwege, um statt reiner Anwendungsdaten IP-Pakete und sogar Ethernet-Frames sowie alle darüber liegenden Protokolle zu tunneln. OpenVPN nutzt für die Datenübertragung bevorzugt das zustandlose UDP, das keine Flusskontrolle für den Datenverkehr kennt. Der Grund: Beim Einkapseln des von den meisten Anwendungen benutzten TCP in andere TCP-Pakete entstehen durch die ineinander geschachtelten TCP-Flusskontrollalgorithmen Interferenzen. In der Folge kann es zu hoher Latenz und Verbindungsabbrüchen kommen.

Da TLS per Definition nur über TCP funktioniert, muss OpenVPN tricksen und gaukelt TLS einfach einen zuverlässigen TCP-Layer vor. Steht der Tunnel, verschlüsselt OpenVPN alle über sein tun/tap-Interface ankommenden Pakete, schreibt einen Initialisierungsvektor davor, authentisiert das Ganze per HMAC und verschickt es über den UDP-Tunnel. Auf der Empfängerseite arbeitet OpenVPN alle Schritte in umgekehrter Reihenfolge ab und leitet das Paket weiter. Bei TLS wird der gesamte Verkehr über einen einzigen UDP-Port abgewickelt. Dazu multiplext der Server sowohl den TLS-Handshake als auch die verschlüsselten Pakete auf eine Verbindung.

Da OpenVPN weder die IP-Adresse noch die UDP-Portnummer des Paketes authentisiert, bereiten NAT-Router auf dem Weg zum Empfänger keine Probleme. Auch Road Warrior mit dynamischen IP-Adressen bedient OpenVPN klaglos. Selbst der Server darf eine dynamische IP-Adresse haben. Dem Client reicht beispielsweise die Angabe eines DynDNS-Namens, um den Server zu erreichen. OpenVPN eignet sich obendrein auch zur Kopplung entfernter Netze.

Tunnel nach Redmonder Art

Das hauptsächlich unter Windows eingesetzte Point-to-Point Tunneling Protocol (PPTP) ist ein VPN-Verfahren für Remote Access und setzt eine verschlüsselte PPP-Brücke auf. Das Point-to-Point-Protokoll (PPP) dürfte den meisten Anwendern geläufig sein: Es sorgt für den Verbindungsaufbau des heimischen PC und die Datenübertragung über Modem- und ISDN-Wählzugängen zum Internet-Provider. Grundsätzlich kann PPP beliebige Protokolle aus höheren Schichten transportieren, etwa IP, IPX, NetBIOS und Appletalk. Bei der Einwahl ins Internet funktioniert die Weiterleitung über den Network Access Server (NAS) aber nur mit IP-Paketen. Um auch andere Protokolle, etwa für Remote Access ins Firmennetz zu transportieren, muss man die PPP-Verbindung quasi über den NAS des Providers zum NAS der eigenen Firma verlängern. Dazu verpackt PPTP die PPP-Pakete über das Tunnel-Verfahren Generic Routing Encapsulation (GRE) in IP-Pakete. Die Signalisierung zum Verbindungsauf- und -abbau erfolgt zwischen dem PPTP-Client und dem Server über eine Control Connection über den TCP-Port 1723.

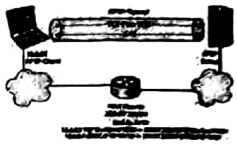
PPTP kapselt die verschlüsselten PPP-Pakete in das GRE-Protokoll, das wie IPSec ein eigenständiges Protokoll ist und keine Ports besitzt.

PPP übernimmt mehrere Aufgaben: Es ist für die Authentifizierung, die Aushandlung der Paketgrößen und IP-Adressen sowie für die Verschlüsselung der Daten zuständig. Die Authentifizierung und die Aushandlung der Schlüssel sind die Schwachpunkte von Microsofts PPTP-Implementierung. Die Authentifizierung beruht auf einem Challenge-Response-Verfahren (MSCHAPv1), bei der der Windows-Client eine vom Server im Klartext verschickte Challenge mit einem Passwort verschlüsselt und an den Server zurückschickt.

Dieser 24 Byte lange LM-Hash lässt sich wie schon die PSKs von IPSec im Aggressive Mode mit Wörterbuch- oder Brute-Force-Angriffen mehr oder minder schnell knacken [3 [4]]. Microsoft hat darauf in MSCHAPv2 mit zahlreichen Verbesserungen reagiert, bei dem der Client unter anderem nicht mehr die ursprüngliche Server-Challenge verschlüsselt, sondern eine davon abgeleitete. Somit ist es erheblich schwieriger, den Hash zu berechnen. Allerdings demonstrierte bereits 2001 eine Gruppe von Studenten, wie man auch MSCHAPv2 innerhalb von Stunden knackt.

Die Datenverschlüsselung von PPTP mit dem Stromchiffrierer RC4 hat ebenfalls ihre Haken: Die in Microsofts Point-to-Point Encryption Protocol (MPPE) eingesetzten Schlüssel beruhen auf dem Passwort, eine dedizierte Schlüsselaushandlung findet nicht statt. In der ersten MPPE-Version gab es sogar nur einen Schlüssel zum Ver- und Entschlüsseln, seit MSCHAPv2 errechnet MPPE für jede Richtung einen eigenen Key. Die Schlüssellänge kann 40 oder 128 Bit betragen. Eine zusätzliche Authentisierung der Pakete nimmt PPTP nicht vor.

Um die Sicherheitsprobleme von PPTP zu beseitigen, hat Microsoft ab Windows 2000 das Extensible Authentication-Protocol (EAP) eingeführt, das die Authentifizierung über Zertifikate und MD5-Hashes unterstützt. Ohne EAP gilt PPTP heutzutage als nicht mehr hinreichend sicher.



Auch PPTP hat unter Umständen Probleme mit NAT-Router. Das Passthrough-Verfahren ebnet diese Hürde.

Neben den Sicherheitsproblemen hat PPTP auch mit NAT zu kämpfen. Manche Router nehmen GRE-Pakete nicht an und verwerfen sie kommentarlos. Zudem gibt es bei GRE keine Ports, sodass die Zuordnung von Ports zu Client-Verbindungen unmöglich ist. Anders als bei IPSec befördert PPTP die zugehörigen Kenndaten unverschlüsselt. Mit PPTP Passthrough sind moderne NAT-Router daher in der Lage, eine Liste der von Clients verwendeten Call IDs zu führen und zuzuordnen.

Als Alternative baut Microsoft seit Windos 2000 das Protokoll L2TP over IPSec (Layer 2 Tunneling Protocol) ein, das wie PPP andere Protokolle höherer Schichten transportiert. L2TP hat keine eigenen Funktionen zur Verschlüsselung. Daher wird es zum Schutz der Daten in Kombination mit IPSec eingesetzt, wobei IPSec dann auch die Authentifizierung übernimmt.

Fazit

IPSec gilt nach wie vor als sicherste VPN-Lösung für alle Szenarien, legt dem Anwender aber einige Steine in den Weg. Unter Windows ist etwa IPSec mit den Bordmitteln nur schwer zu konfigurieren. Nach und nach finden Erweiterungen und Verbesserungen Eingang in den Standard, um viele der Hürden zu beseitigen. Im Dezember 2005 wurde die zweite Generation von IPSec Standards (RFC 4301Ð4309) veröffentlicht. Auch das schnellere und weniger kompliziertere IKEv2 soll im nächstem Jahr IKEv1 ablösen.

IPSec-VPNs sind relativ ressourcenhungrig und erzeugen viel Overhead, allerdings sind sie skalierbar. Unter anderem verhelfen spezielle Kryptobeschleuniger den VPN-Gateways zu noch mehr Leistung, um viele parallele Verbindungen bedienen zu können. Microsofts PPTP ist in fast jedem Windows-Client verbaut, bietet aber nur mit EAP ausreichend Schutz vor Angriffen. Was bleibt, ist OpenVPN, das sich als echter Shooting Star entpuppt. Aufgrund seiner Flexiblität und der hohen Sicherheit ist damit zu rechnen, dass OpenVPN bald sehr viele Anhänger finden wird. Seine Leistungsfähigkeit gegenüber IPSec muss es aber erst noch beweisen, denn durch das Hin- und Herkopieren der Daten zwischen verschiedenen Schnittstellen wird einiges an Zeit vergeudet. Die einfache Installation, Konfiguration und Robustheit machen derzeit die Wahl fürs Open-Source-VPN dennoch leicht.

Literatur

[1] Michael Thumann: **Einbruch ins VPN, Nachlässige Konfigurationen führen zu unsicheren VPNs [5]**, Hintergrundartikel auf heise Security

[2] Stephan Scholz, Johannes Endres, Das Überall-VPN, Mit dem Browser sicher ins LAN, c't 15/04, S. 194

[3] Daniel Bachfeld, **Mit roher Gewalt, Angriff auf Passwörter in Windows-Netzwerken [6]**, Hintergrundartikel auf heise Security

---

Authentifizierung

Der sichere Betrieb von VPNs steht und fällt mit einer guten Authentifizierung. State-of-the-Art sind X.509v3-Zertifikate, in der die Kopplung eines öffentlichen Schlüssels an eine Identität durch eine Certification Authority (CA) beglaubigt ist. Vertraut man dieser CA, kann man sicherstellen, dass nur bekannte Identitäten Zugriff auf das VPN haben. Im Gegenzug können die Teilnehmer verifizieren, ob sie auch wirklich mit dem gewünschten VPN-Server eine Verbindung aufgenommen haben. Eine Identität kann sowohl eine Person als auch ein PC darstellen.

Der Einsatz von Zertifikaten bedeutet aber einen höheren Verwaltungsaufwand, da eine Public-Key-Infrastrukur (PKI) vonnöten ist, die Zertifikatsanträge bearbeitet, Zertifikate ausstellt und verteilt sowie Listen über ungültige beziehungsweise zurückgezogene Zertifikate führt und zur Verfügung stellt (Certificate Revocation List, CRL). Dazu ist nicht unbedingt ein externer Dienstleister erforderlich. Steht ein Windows 2003 Server zur

Verfügung, so kann dessen leistungsfähige CA zum Ausstellen von Zertifikaten verwendet werden. Der Administrator kann eine eigene vollständige PKI aber auch mit kostenlosen Open-Source-Lösungen wie OpenCA aufbauen, die viele Schritte automatisiert durchführt.

Weniger automatisch, dafür einfacher in der Installation und Konfiguration sind Lösungen wie TinyCA, deren Zertifikatsverwaltung komplett über eine GUI bedient wird. OpenVPN liefert gleich seine eigene CA Easy-RSA mit. Mit simplen Skriptaufrufen erstellt der Nutzer ein CA-Zertifikat und unterschreibt damit Server- und Client-Schlüssel. Bei sehr vielen Teilnehmern kommen derart simple Lösungen aber schnell an ihre Grenzen.

VPN-Lösungen können zur Authentifizierung statt eines Zertifikates auch den unsignierten öffentlichen Schlüssel des Gegenübers nutzen. Der muss aber, ähnlich wie bei PSKs, vorher auf einem separaten Weg ausgetauscht werden. Auf den Einsatz von PSK sollte man am besten ganz verzichten oder wenigstens sehr schwer erratbare Passwörter wählen.

Zweiter Grenzposten

Einige Herstellerimplementierungen von IKE kennen von Haus aus nur die Geräte-Authentifizierung, die nicht besonders abgestuft ist. Um eine Authentifizierung auf Nutzerebene zu ermöglichen und die Schwächen des IKE Aggressive Modus beim Gebrauch von PSKs auszubessern, unterstützen einige Hersteller das XAUTH-Verfahren. Dabei wird IKE um eine Phase erweitert, in der Mechanismen wie RADIUS, SecurID und andere zum Einsatz kommen. Erst wenn beide Authentifizierungen erfolgreich waren, darf der Nutzer mit seinem PC ins VPN.

VPN-Glossar

Der **Advanced Encryption Standard** (AES) ist der Nachfolgeverschlüsselungsstandard von DES (Data Encryption System). **3DES** mit 128 Bit gilt zwar immer noch als sicher, ist aber wegen der Dreifachverschlüsselung um Faktoren langsamer als AES. AES unterstützt 128, 192 und 256 Bit lange Schlüssel.

Anders als der **ECB-Mode** (Electronic Code Book) verhindert der **CBC-Mode** (Cipher Block Chaining) bei Verschlüsselungsalgorithmen die Entstehung von Mustern im Chiffrat, die Rückschlüsse auf den Klartext zulassen. Dazu lässt CBC das Ergebnis der vorherigen Blockoperation in die aktuelle einfließen (Chaining). Sowohl ECB als auch CBC sind für so genannte Bit-Flipping-Attacken anfällig. Dabei kann ein Angreifer im Chiffrat ohne Kenntnis des Schlüssels einzelne Bits manipulieren, ohne später beim Entschlüsseln durch den Empfänger einen Fehler zu provozieren. Da sich so der Inhalt manipulieren lässt, muss auch für verschlüsselte Pakete die Integrität gesichert werden, etwa mit HMAC.

In CBC werden jeweils Blöcke von jeweils 16 Datenbytes verschlüsselt. Das CBC-Verfahren wird mit einem zufällig gewählten 128 Bit langen Initialisierungsvektor initialisiert, der etwa bei ESP-Paket den chiffrierten Nutzdaten vorangestellt wird. Da die Daten blockweise verschlüsselt werden, muss der letzte Datenblock mit Füll-Bytes zur vollen Blocklänge aufgefüllt und die Anzahl dieser Stopf-Bytes in einem Längen-Byte festgehalten werden.

Über das **Diffie-Hellman-Key-Exchange-Verfahren** (DH) lassen sich kryptographische Schlüssel sicher über unsichere Kanäle aushandeln. Es ist selbst kein Verschlüsselungsverfahren und tauscht auch keine Schlüssel im eigentlichen Sinne aus. Das von Martin Hellman und Whitfield Diffie entwickelte Verfahren beruht auf den Eigenschaften diskreter Logarithmen: zwar ist es einfach, eine Zahl zu potenzieren. Es ist aber nur mit sehr großem Aufwand möglich, den diskreten Logarithmus einer Zahl zu berechnen. Bei der Aushandlung einigen sich die VPN-Peers auf eine Primzahl p und eine Primitivwurzel g mod p. Beide Faktoren dürfen unverschlüsselt übertragen werden. Anschließend erzeugt jede Seite eine geheime Zufallszahl a/b und berechnet daraus den Wert Za= ga mod p beziehungsweise Zb = gb mod p. Za und Zb werden an den Partner übertragen.

Daraus kann nun jede Seite den gemeinsamen symmetrischen Schlüssel K berechnen: Zba mod p = Zab mod p = K. Sind die eingesetzten Zahlen hinreichend groß, ist es für einen Angreifer so gut wie unmöglich, den Key zu knacken. Große Zahlen erfordern allerdings mehr Rechenaufwand. Die Größe der Zahlen bestimmt die gewählte DH-Gruppe. Die kleinste **DH Gruppe** 1 hat 768 Bits und die größte definierte Gruppe 18 besitzt 8192 Bits. Empfohlen wird derzeit der Gebrauch der Gruppe 5 mit 1536 Bits.

Über einen **Hash-Algorithmus** lässt sich aus einem beliebig langen Datensatz eine Prüfsumme fester Länge berechnen. Dieser Hashwert soll möglichst einmalig sein, man spricht dann auch von Kollisionsfreiheit. Damit ist sichergestellt, dass der Datensatz nicht so manipuliert werden kann, dass der Hashwert trotzdem noch derselbe ist. Üblicherweise kommen **SHA-1** (Secure Hash Algorithm) mit 160 Bits und **MD5** (Message Digest Algorithm) mit 128 Bits zum Einsatz. Bei ESP wird der Hashwert von 128, respektive 160 Bit auf 96 Bit abgeschnitten. Zur Authentisierung von Daten dienen (Keyed-)Hash Message Authentication Codes. **HMAC** ist eine Sonderform des MAC, bei der zusammen mit einem geheimen Schlüssel ein Hash-Wert etwa über Datenpaket gebildet wird. Bei VPNs benutzt man in der Regel **HMAC-MD5** oder **HMAC-SHA-1**. Unter IKE sorgt die Aktivierung der Funktion **Perfect Forward Secrecy** (PFS) für frisches Schlüsselmaterial bei der Aushandlung eines Keys für den Quick Mode. Ohne PFS leitet Phase 2 die ESP Schlüssel vom DH-Geheimnis der IKE Phase 1 ab. Wird etwa im IKE Main Mode nur alle 24 Stunden authentisiert, so hängen alle in diesem Zeitraum erstellten IPsec SAs von diesem Master-Schlüssel ab. Knackt ein Angreifer den Schlüssel, so wären alle Sessions eines Tages kompromittiert.

Mit einem **Proposal** signalisiert ein IPSec-VPN-Peer, mit welchen Algorithmen er umgehen kann und welche DH-Gruppe er verwenden will. Zur Verschlüsselung implementieren die meisten Hersteller 3DES und AES mit verschiedenen Schlüssellängen jeweils im CBC-Mode. Für die Authentisierung sorgen die Hash-Algorithmen SHA-1 und MD5.

Anhand des **Security Parameters Index** (SPI) kann ein IPSec-Peer die zum Paket gehörige **Security Association** (SA) erkennen, um es zu entschlüsseln und die Authentizität zu prüfen. In den Security Associations (SA) speichern die Peers die Konfiguration einer VPN-Verbindung, etwa die benutzten kryptographischen Algorithmen, die Zeit bis zur Neuberechnung von Schlüsseln und so weiter. Phase 1 und Phase 2 besitzen jeweils eigene SAs, die nur unidirektional gelten. Pro Phase benötigt ein Peer also jeweils eine SA für eingehenden und ausgehenden Datenverkehr.

URL dieses Artikels:
http://www.heise.de/security/artikel/VPN-Knigge-270796.html

Links in diesem Artikel:
[1] http://www.heise.de/security/artikel/VPN-Glossar-271454.html#glossar
[2] http://www.heise.de/security/artikel/Fazit-271450.html#uellt
[3] http://www.heise.de/security/artikel/Fazit-271450.html#uellt
[4] http://www.heise.de/security/artikel/Fazit-271450.html#uellt
[5] http://www.heise.de/security/artikel/Einbruch-ins-VPN-270592.html