Windows 2000 Sicherheitskonzepte

Skript zum Vortrag vom 14.02.2001 an der Hochschule Rapperswil
von Marcel Liebi

Inhaltsverzeichnis

1 Einleitung

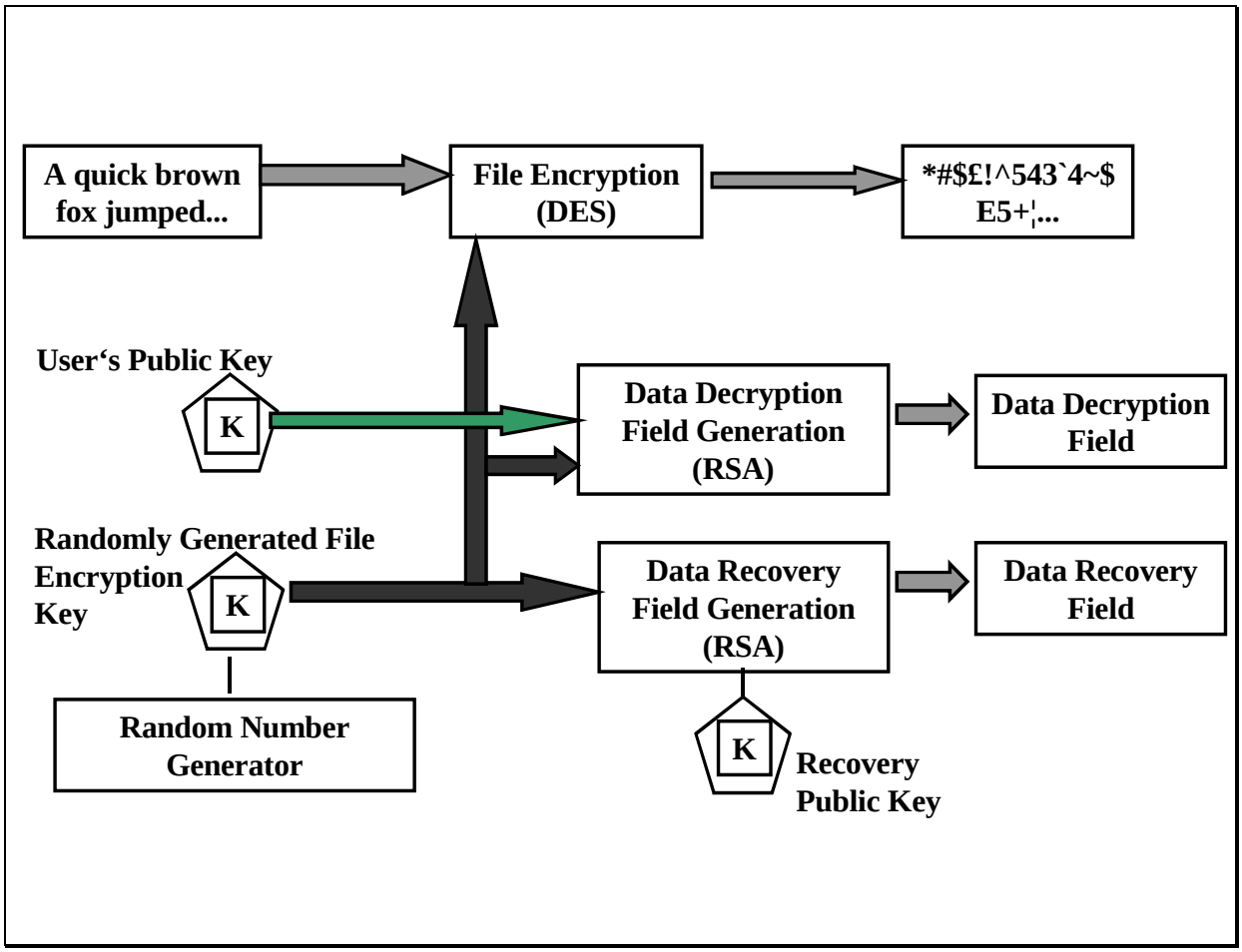
Seit Februar 2000 ist unter der Bezeichnung Windows 2000 ein Betriebssystem auf dem Markt, das laut Microsoft die bestehenden Sicherheitskonzepte weit in den Schatten stellt. Ich werde auf den kommenden Seiten auf die Neuerungen von Windows 2000 im Bezug auf Sicherheit eingehen. Schwergewichtig wird das Kerberos-Verschlüsselungskonzept, der Login-Prozess und die Verschlüsselung von Dateien erklärt.

2 EFS (Encrypted File System)

Ein grosser Schritt Richtung Datensicherheit gerade auf tragbaren Geräten ist das Encrypted File System (EFS). Bei Verlust eines Notebooks stellt sich sehr oft die Frage, wie sicher die Daten vor dem Zugriff durch Dritte sind. Selbst unter Windows NT konnte noch auf die Daten zugegriffen werden, wenn man die Festplatte ausbaute und in ein bestehendes System einbaute, in dem man Administratorrechte hatte, oder nicht das lokale Betriebssystem startete.
Dies ist mit EFS nicht mehr möglich. Unter Windows 2000 wird für jeden Benutzer ein Private- und Public-Key mit der Kryptographiestärke 128 bit (ähnlich dem bekannten PGP-Verfahren) erzeugt, deren Anwendung vollständig im Betriebssystem gekapselt ist und für den Benutzer ohne weitere Massnahmen zur Verfügung steht. Mit deren Hilfe verschlüsselt Windows 2000 alle lokalen Daten auf der Festplatte, wenn diese das Encrypted-Attribut besitzen. Nur noch der Besitzer kann dann die so verschlüsselten Daten entschlüsseln, selbst der Administrator kann dies nicht mehr.
Als Sicherheitsmassnahme ist allerdings die Wiederherstellung der Daten mittels dem lokalen Betriebssystem und einem sogenannten Desaster Recovery Agent Key möglich, falls der Benutzer nicht mehr erreichbar, oder gelöscht ist. Aber selbst dann benötigt man das aktuelle lokale Betriebssystem mit Administratorrechten. Eine Neuinstallation von Windows 2000 auf den Rechner würde auch nichts bringen, da beim Erzeugen der Benutzer wieder neue Keys erzeugt werden.

2.1 Verschlüsselung (encryption)

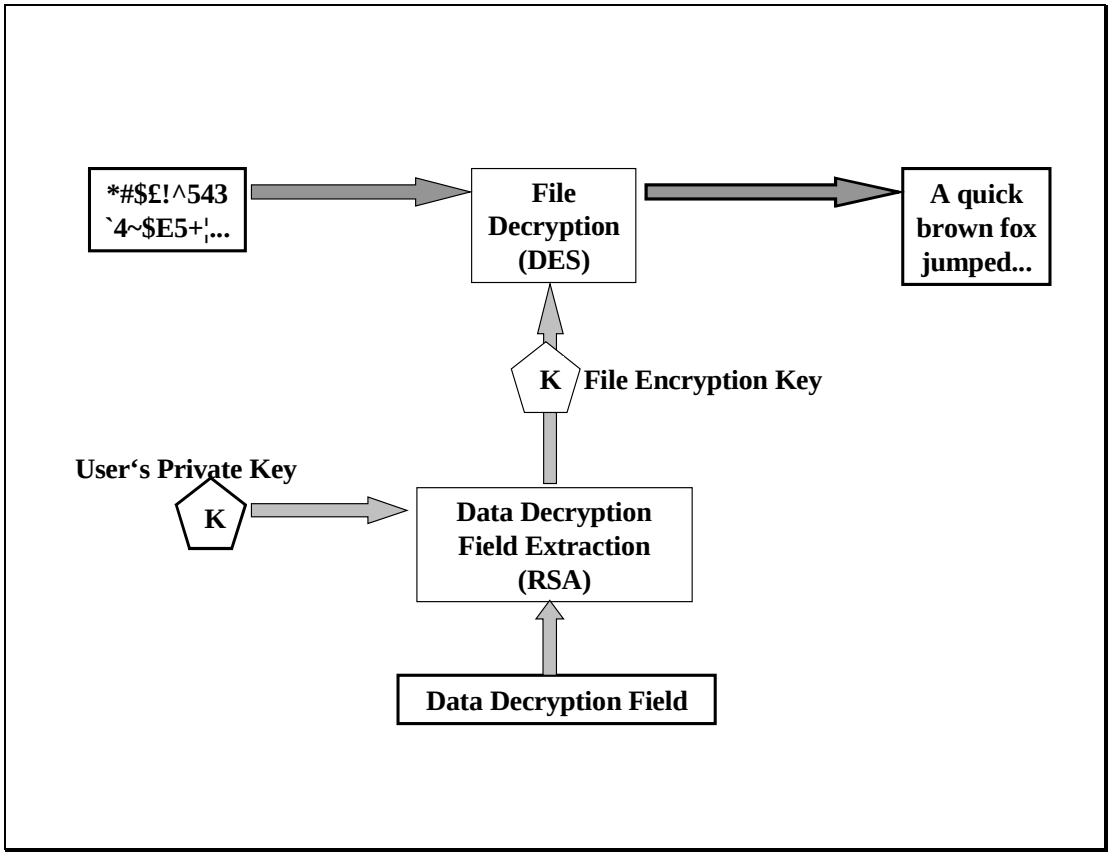
Grundsätzlich basiert das EFS auf einem Hybridverfahren: Hierbei gelangen mehrere Verschlüsselungsverfahren nacheinander zum Einsatz, das heisst zusätzlich zu einer symmetrischen auch eine Chiffrierung mit öffentlichen/privaten Schlüsseln.
Zunächst wird eine Datei mit Hilfe eines DES-Algorithmus – mittlerweile auch außerhalb der USA mit 120 bit – symmetrisch verschlüsselt. Der dazu verwendete Schlüssel, im englischen „File Encryption Key“ (FEK) genannt, wird vom Betriebssystem zufällig generiert. Zusätzlich verschlüsselt das EFS den FEK mit dem öffentlichen Schlüssel aus dem öffentlichen/privaten Schlüsselpaar des Anwenders (respektive den öffentlichen Schlüsseln mehrerer Anwender). Diese nun chiffrierten FEKs werden mitsamt der Datei als spezielles EFS-Attribut im „Data Decryption Field“ (DDF) abgelegt. Als Ergebnis besitzt man eine Datei, die symmetrisch per DES verschlüsselt worden ist. Der dabei zugrundegelegte Schlüssel wurde anschließend auf Basis eines öffentlichen/privaten Schlüsselverfahrens mit den entsprechenden öffentlichen Schlüsseln abermals chiffriert. Nur die so zustande kommenden Schlüsselinformationen werden zusammen mit der Datei gespeichert, denn ohne den zugehörigen privaten Schlüssel ist keine Wiederherstellung der in ihr enthaltenen Informationen möglich. Die entsprechenden privaten Schlüssel wiederum werden bei dem jeweiligen Anwender oder dem Recovery-Agenten abgelegt – etwa im Active Directory, auf Smartcards oder in anderen gesicherten Bereichen.
Zur eventuellen Datenwiederherstellung durch einen Administrator werden die FEKs zudem mit einem oder mehreren öffentlichen Recovery-Schlüsseln verschlüsselt und im „Data Recovery Field“ (DRF) ebenfalls als EFS-Attribut bei der Datei gespeichert. Entsprechende Mechanismen hat Microsoft in das EFS eingebaut, um Unternehmen für Notfälle, das heisst wenn ein Mitarbeiter nicht mehr erreichbar sein sollte, die zwangsweise Entschlüsselung zu eröffnen: Verlässt zum Beispiel ein Anwender die Firma oder verliert er seinen privaten Schlüssel, besteht somit die Möglichkeit, den Entschlüsselungsvorgang über die privaten Recovery-Agent-Schlüssel durchzuführen.



Schematischer Ablauf der Verschlüsselung

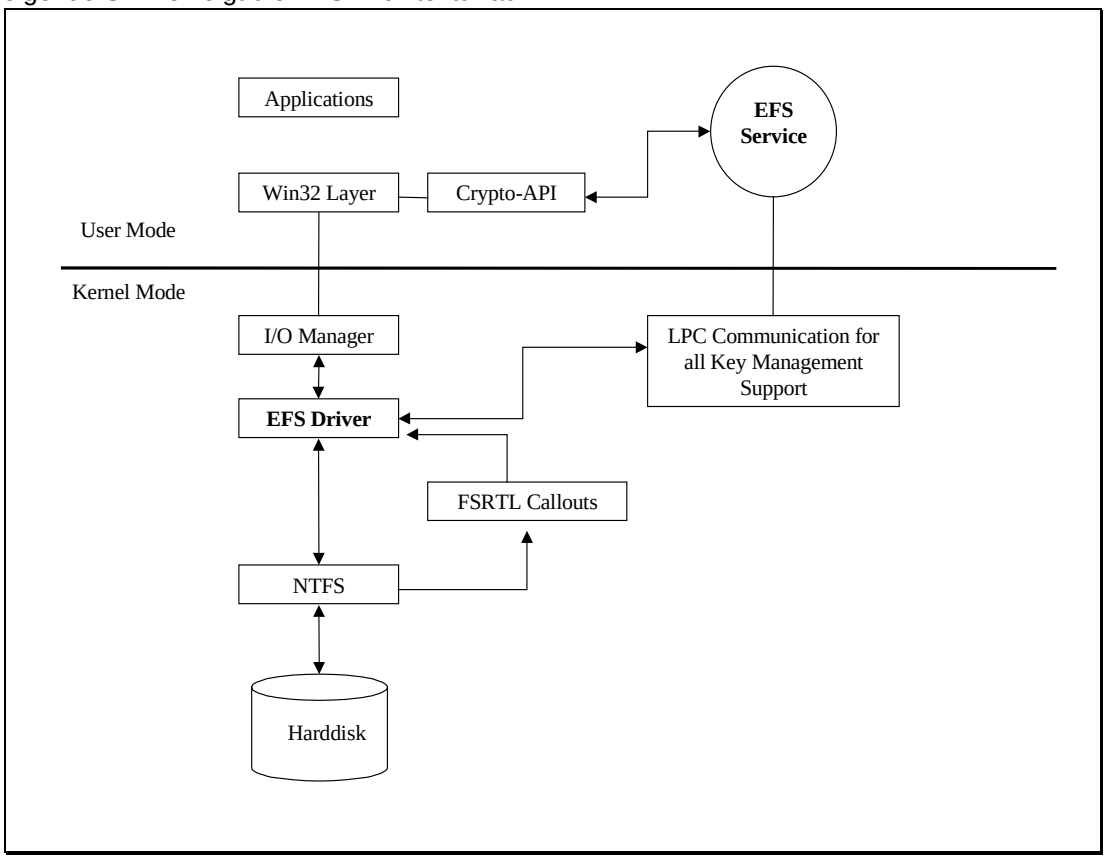
2.2 Entschlüsselung (decryption)

Die Entschlüsselung funktioniert folgerichtig genau auf umgekehrtem Weg: (siehe Bild 2): Nur mit Hilfe des privaten Schlüssels des Anwenders kann der FEK wieder hergestellt werden. Mit dem so gewonnenen FEK lässt sich sodann eine Dechiffrierung der Datei durchführen. Das Besondere daran: Alle diese Vorgänge laufen unsichtbar im Hintergrund ab. Der Anwender wird nicht mit störenden Unterbrechungen, wie etwa die Aufforderung zur Eingabe von Kennwörtern, konfrontiert. Die direkte Integration in den Windows-Explorer gestattet eine einfache Nutzung der Datenverschlüsselungs-funktion. Das Versehen des entsprechenden Kontrollkästchens mit einer Markierung reicht aus, um einen Ordner oder eine einzelne Datei von Windows 2000 verschlüsseln zu lassen (siehe Bild 3). Alternativ steht auf Betriebssystemebene der Befehl CIPHER zur Verfügung.
Wann immer es um die Kryptographie in Verbindung mit Schlüsseln geht, stellt sich die Frage nach der Stärke der Verschlüsselung – je größer ein Schlüssel ist, desto schwerer fällt es einem Angreifer, den Code zu knacken. Durch die Lockerung der Exportbestimmungen ist jetzt auch außerhalb der USA die Verwendung von 120 bit-Schlüsseln möglich geworden.



Schematischer Ablauf der Enschlüsselung

Die folgende Skizze zeigt die EFS-Architektur auf.



Für Microsoft spricht, dass EFS im Kernel-Modus eingebaut ist und das I/O System darauf aufbaut. Es besteht also keinerlei Möglichkeit, Daten unberechtigterweise zu übernehmen.

3 Kerberos

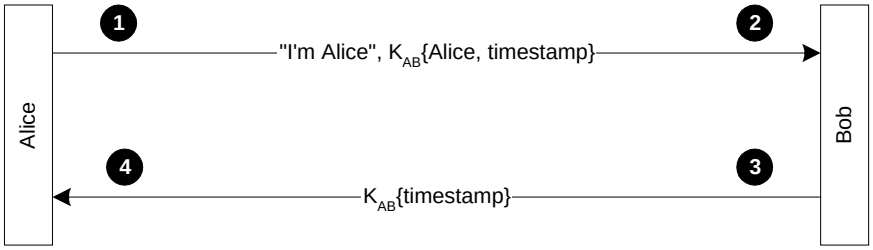
Kerberos wird häufig als wichtigste Verbesserung im Bereich der Sicherheit genannt. Von wenigen Stellen abgesehen, sieht auch der Administrator von Windows 2000 nichts von Kerberos. Das Protokoll arbeitet im Hintergrund und ist für die Sicherheit bei der Authentifizierung von Clients und Servern verantwortlich. Gerade deshalb wird hier Kerberos genauer erklärt.
Nicht für die Authentifizierung von Personen, sondern für die Bestimmung der Authentizität von Anfragen von Netzwerkressourcen dient der Kerberos-Algorithmus. Er wurde im Rahmen des Project Athena am Massachusetts Institute of Technology (MIT) entwickelt. Es ist hiermit möglich, eine ‚real-time' Authentifizierung in einem verteilten System durchzuführen.

Kerberos beseitigt unauffällig aber effektiv eine Reihe von Schwächen, die insbesondere der LM-Teil (LAN-Manager) der NTLM-Authentifizierung aufweist. Im Vergleich mit NTLMv2, der aktuellsten Version der NTLM-Authentifizierung, ist entscheidend, dass Server bei Kerberos die Identität des Benutzers annehmen können. Damit kann beispielsweise ein Web Application Server eine Authentifizierung im Kontext des Benutzers, der sich bei ihm angemeldet hat, am Datenbankserver vornehmen. Das bringt nicht nur höhere Sicherheit, sondern erleichtert auch die Realisierung von Sicherheitskonzepten in solchen Anwendungen, die heute oft mit ‚selbstgestrickten' Sicherheitsmodellen arbeiten.

3.1 Grundlegendes Konzept

Die Frage ist, wie ich wissen kann, ob mein Partner auch der ist, den er angibt zu sein? Das Konzept ist einfach: Beide Parteien haben ein gemeinsames geheimes Passwort, das nur sie wissen. Die Identität des anderen kann nun verifiziert werden, indem dieser beweist, dass er das Passwort kennt. Doch wie teilt man einander das Passwort in einem öffentlichen Netz mit, so dass niemand anders das Passwort erfährt? Das Kerberosprotokoll bringt hier die Lösung mit einer Private-Key-Cryptography. Anstelle eines gemeinsamen Passwortes haben beide Partner einen symmetrischen Schlüssel. Wie jedes Protokoll braucht auch Kerberos einen sogenannten Authenticator, das heisst, Informationen, die die Identifikation des Partners erlauben.
Wie das Ganze wirklich funktioniert, erkennt man am Besten an einem Beispiel.
Angenommen Alice möchte Bob etwas senden, weiss aber nicht sicher, ob der Zielrechner wirklich Bob ist.
Alice und Bob haben bereits den symmetrischen Schlüssel, den Session Key erhalten.

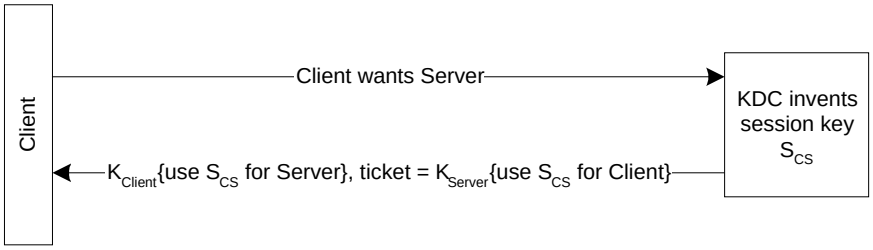
1. Alice sendet also eine Nachricht an Bob mit ihrem Namen im Klartext und einem mit dem Session Key verschlüsselten Authenticator. Der Authenticator ist eine Struktur mit zwei Feldern. Ein Feld enthält Informationen über Alice. Beispielsweise nochmals der Name von Alice. Das zweite Feld enthält die aktuelle Zeit von Alice's Rechner.
2. Bob erhält die Nachricht und sieht, dass sie von jemandem ist, die sich Alice nennt. Bob verwendet nun den Session Key für die Sitzung mit Alice, um den Authenticator zu entschlüsseln. Gelingt das, liest er die Zeit aus dem Authenticator aus und prüft durch einen Vergleich mit seiner eigenen Zeit, ob das Paket angenommen werden darf. Die Sicherheitseinstellungen des Systems legen dieses Zeitfenster fest. Pakete die mit einer bereits erhaltenen oder zu lange vergangenen Zeit ankommen, werden weggeworfen.
3. Ist das Paket gültig, sendet Bob die von Alice erhaltene Zeit mit dem Session Key verschlüsselt an Alice zurück.
   Hier ist zu beachten, dass Bob nicht den gesamten Authenticator an Alice zurücksendet, sondern nur einen Teil davon. Nur so ist es Alice möglich, festzustellen dass auch wirklich Bob ihr Gesprächspartner ist.
4. Alice erhält die Antwort von Bob, entschlüsselt diese Antwort und vergleicht die Zeit. Stimmt die erhaltene und die gesendete Zeit überein, weiss Alice dass wirklich Bob auf der anderen Seite ist.



Figur 1: Mutual authentication (Alice-Bob) Quelle: Whitepaper von Microsoft, Windows 2000 Kerberos Authentication

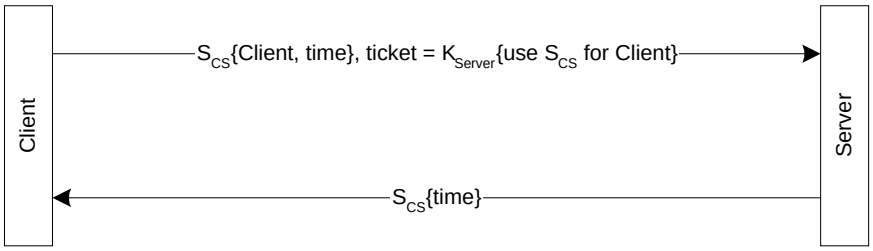
3.2 Key Distribution

Im obigen Beispiel wurde vorausgesetzt, dass Alice und Bob bereits einen Session Key besitzen. In der Realität kann jedoch nicht davon ausgegangen werden. Zuerst muss dieser Schlüssel an Alice und Bob verteilt werden. Dazu ist eine dritte Instanz notwendig, der vertraut werden kann und die die Authentifizierung vornimmt. In Windows 2000 ist das der Kerberos Server, das sogenannte Key Distribution Center (KDC). Dieser Server verwaltet alle Schlüssel für die Benutzer seiner Domäne, erzeugt die Session Keys und beglaubigt die Identität eines dienstanfordernden Benutzers gegenüber dem Dienstanbieter. Die Schlüssel, die der Server verwaltet, werden für den Austausch des Session Keys benötigt. Sie werden mit einem Algorithmus aus dem Benutzerpasswort abgeleitet.
Wenn ein Client mit einem Server sprechen möchte, sendet er zuerst eine Anfrage an das KDC. Das KDC generiert darauf einmaligen symmetrischen Session Key und sendet eine Antwort an den Client zurück. Diese Anwort besteht aus zwei Teilen. Der erste Teil ist der Session Key, verschlüsselt mit dem Schlüssel des Clients(KDC/Client-Schlüssel), der zweite Teil ist der Session Key und Informationen über den Client, verschlüsselt mit dem Schlüssel des Servers (KDC/Server-Schlüssel). Dieser zweite Teil wird auch Session Ticket genannt. Da das KDC das Passwort des Servers kennt, kann es auch den entsprechenden Schlüssel generieren um diesen zweiten Teil zu verschlüsseln.



Figur 2: Key Distribution (in practice). Quelle: Whitepaper von Microsoft, Windows 2000 Kerberos Authentication

Man sieht, dass das KDC nur ein Ticket-Verteil-Center ist und nicht für die richtige Ankunft der Schlüssel garantiert. Dies macht aber auch nichts, denn ein anderer Empfänger könnte mit der Nachricht nichts anfangen, da er sie nicht entschlüsseln kann.
Wenn der Client die Nachricht empfängt, entschlüsselt er den Session Key und speichert ihn mit dem Ticket in einem sicheren Bereich des RAM. Will der Client nun Verbindung zum Server aufbauen, sendet er ihm eine Nachricht mit einem Authenticator, der mit dem Session Key verschlüsselt ist und mit dem Ticket, das immer noch vom KDC her verschlüsselt ist. Der Server kann nun dieses Ticket entschlüsseln, denn er hat ja den passenden Schlüssel. Er entnimmt dem Ticket die Informationen über den Client und den Session Key. Mit dem Session Key entschlüsselt er den Authenticator und sendet, wie im obigen Beispiel, die empfangene Zeit mit dem Session Key verschlüsselt zurück. Der Server müsste den Session Key nicht speichern, da er bei jeder Anfrage vom Client ein Ticket mit diesem Session Key erhalten könnte.



Figur 3: Mutual authentication (Client/server). Quelle: Whitepaper von Microsoft, Windows 2000 Kerberos Authentication

Damit die Tickets nicht gestohlen und wiederverwendet werden können, haben sie ein Ablaufdatum, das vom KDC festgelegt wird.

3.3 Ticket-Granting Ticket

Arbeitet man schon mit Tickets, dann braucht der Client auch ein Ticket für das KDC. Der Client/KDC-Schlüssel ist wie schon erwähnt vom Passwort des Benutzers abgeleitet. Wenn Alice einloggt, akzeptiert der Kerberos-Client ihr Passwort und konvertiert dieses in einen kryptografischen Schlüssel indem das Passwort eine Hash-Funktion durchläuft. (Microsoft verwendet DES-CBC-MD5). Das Resultat ist der Client/KDC-Schlüssel. Das KDC hat seine Kopie des Schlüssels aus der Benutzerdatenbank. Erhält das KDC nun eine Anfrage des Clients, so kann das KDC den Schlüssel erstellen und die Anfrage bearbeiten. Dieser Prozess (berechnen des Schlüssels) wird nur einmal während einer Session durchgeführt. Das KDC erstellt nämlich ein Ticket für sich selber und übermittelt dieses Ticket, verschlüsselt mit dem Client/KDC-Schlüssel, dem anfragenden Client. Dieses Ticket wird Ticket-Grant Ticket(TGT) genannt. Der Client kann nun mit diesem Ticket das KDC kontaktieren wie einen anderen Server. Das Passwort wird von nun an auch nicht mehr gebraucht.

3.4 Unterprotokolle

Das Kerberosprotokoll besteht aus drei Unterprotokollen. Das Unterprotokoll in welchem das KDC dem Client mit dem Client/KDC-Schlüssel ein TGT übergibt, nennt man Authentication Service (AS). Das Unterprotokoll in welchem das KDC den Session Key und das Session Ticket dem Client übermittelt, nennt man Ticket-Granting Service (TGS) Exchange. Das Unterprotokoll mit dem der Client mit dem Ticket einen Service anfordert, nennt man Client/Server (CS) Exchange.

Nochmals ein Beispiel, um die Zusammenarbeit der Unterprotokolle klar zu machen.

AS Exchange

Alice meldet sich beim Netzwerk an. Sie tippt ihren Benutzernamen und ihr Passwort ein. Der Kerberosclient konvertiert ihr Passwort in ein Client/KDC-Schlüssel und speichert das Ergebnis im flüchtigen Speicher. Dann sendet der Client ein Authentication Request an das KDC. Diese Nachricht enthält den Benutzernamen und den Service, der angefordert wird. In diesem Fall der Ticket Granting Service. Der zweite Teil der Nachricht ist der sogenannte Authenticator, der beweisen soll, dass das KDC das Passwort kennt. Wie schon beschrieben, ist dies unter anderem ein Zeitstempel, der mit dem Client/KDC-Schlüssel verschlüsselt ist. Wenn das KDC die Nachricht erhält, sieht es in seiner Datenbank nach dem Passwort von Alice und generiert daraus auch den Client/KDC-Schlüssel. Damit kann der Authenticator entschlüsselt werden und die Identität von Alice ist somit bestätigt. Das KDC prüft, ob Alice den geforderten Service in Anspruch nehmen darf. Dann macht es einen Session Key für die Sitzung des Clients mit dem KDC. Eine Kopie des Schlüssels wird mit dem Client/KDC-Schlüssel verschlüsselt, eine zweite Kopie in ein TGT verpackt. Das TGT wird mit dem privaten Schlüssel des KDC verschlüsselt. Beide Pakete werden dann an den Client zurückgesendet. Der Client entschlüsselt seinen Session Key und speichert ihn zusammen mit dem TGT im flüchtigen Speicher. Den Client/KDC-Schlüssel braucht er von nun an nicht mehr. Den Session Key nennen wir nun Logon Session Key

TGS Exchange

Alice möchte mit Bob kommunizieren. Dazu braucht sie ein Ticket für Bob. Sie sendet deshalb dem KDC einen Ticket Granting Service Request. Dieser Request enthält Alice's Name, einen Authenticator, der mit dem Logon Session Key verschlüsselt ist, sowie das TGT und den Namen des Services, den Alice gerne hätte. Das KDC entschlüsselt das TGT und erhält so den Logon Session Key. Damit kann es den Authenticator entschlüsseln und testen. Das KDC erstellt nun einen neuen Session Key für die Kommunikation von Alice mit Bob. Dieser Session Key wird mit dem Logon Session Key von Alice verschlüsselt. Eine Kopie des Session Keys wird in ein Ticket eingebettet und dieses Ticket mit dem Server/KDC-Schlüssel, den nur Bob und das KDC kennen, verschlüsselt. Beides wird dann an den Client zurückgeschickt. Alice entschlüsselt nun den Session Key und speichert diesen, sowie das Ticket, im Speicher.

CS Exchange

Alice sendet Bob einen Request für einen Service. Dieser Request enthält einen Authenticator, verschlüsselt mit dem Session Key und das eben erhaltene Ticket. Nachdem Bob das Paket erhalten hat, entpackt er das Ticket. Mit dem daraus erhaltenen Session Key entschlüsselt er den Authenticator von Alice und prüft dessen Echtheit. Von jetzt an besteht eine authentifizierte Verbindung zwischen Alice und Bob.

3.4.1 Was passiert wenn ein Ticket abgelaufen ist?

Verwendet der Client ein abgelaufenes Ticket für die Verbindung zu einem Server, so sendet dieser eine Fehlermeldung zurück. Der Client muss dann beim KDC ein neues Ticket anfordern.
Ist das TGT abgelaufen, braucht der Client wieder das Passwort des Benutzers um den Client/KDC-Schlüssel abzuleiten und dann ein neues TGT anzufordern.

3.5 Key Distribution Center (KDC)

Windows 2000 implementiert das KDC als einen Domänenservice. Es braucht das Active Directory der Domäne als Datenbank für die Benutzerinformationen. Das KDC ist ein einziger Prozess, welcher zwei Services zur Verfügung stellt. Es sind dies der Authentication Service (AS) und der Ticket-Granting Service (TGS). Das KDC ist in jedem Domänenkontroller vorhanden, als ein Service des Active Directory.

3.6 Angriffspunkte

Die Integrität des gesamten Systems ist zunichte gemacht, falls der Kerberos-Server erfolgreich angegriffen wird, es also gelingt über ihn in Besitz der geheimen Schlüssel zu gelangen, oder er derart manipuliert wird, dass seine Authentifizierungen unglaubwürdig werden.
Windows 2000 macht ausserdem automatisch die Authentisierung über NTLM falls der Client kein Kerberos kann. Siehe ‚Sicherheitslöcher'.

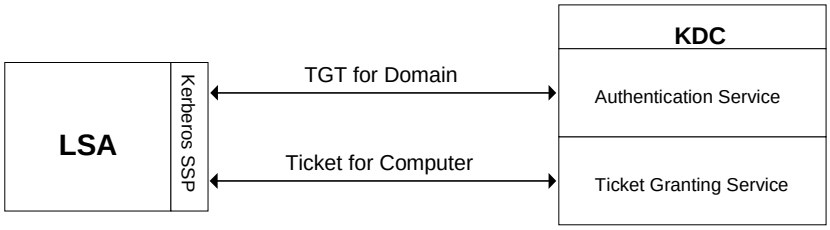
4 Logon Prozess

Möchte ein Benutzer mit einem Account in einer Windows 2000 Domäne einloggen, folgt ein Prozess mit drei Schritten.

1. Der Benutzer (Client) möchte ein Ticket vom KDC. Dies holt er mit einem AS Exchange.
2. Der Benutzer holt sich ein Ticket für die Services auf seinem Computer.
3. Der Benutzer fragt die Berechtigungen am lokalen System ab.

4.1 Passwort Logon

Alice startet den Logon Prozess, indem sie Ctrl+Alt+Del drückt. Es erscheint eine Eingabemaske, in welche sie ihren Benutzernamen und das Passwort eingibt. Diese Daten werden von einem Task an die Local Security Authority (LSA) gegeben, damit die Validierung stattfindet. Das Passwort wird sofort in den privaten Client/KDC-Schlüssel konvertiert.



Figur 4: Interactive logon to a domain account. Quelle: Whitepaper von Microsoft, Windows 2000 Logon Process

Das Betriebssystem kreiert eine sogenannte Window-Station und verschiedene Desktopobjekte und hängt diesen das Zugriffstoken von Alice an. Dieses Token wurde von der LSA kreiert und erlaubt oder verhindert den Zugriff auf die Objekte.

4.2 Smartcard Logon

Man braucht beim Smartcard Logon nicht wie beim Passwort Logon, wo symmetrische Schlüssel verwendet werden, asymmetrische Schlüssel.
Der Logon Prozess beginnt mit dem Einsetzen der Smartcard in den Computer. Dies löst bei Windows 2000 ein Logonereignis aus. Der Benutzer muss nun aber nicht wie bekannt ein Passwort, sondern einen PIN eingeben. Der PIN wird an die LSA gegeben, die diesen benutzt, um auf die Smartcard zuzugreifen. Die Smartcard beinhaltet unter anderem den privaten Teil des asymmetrischen Schlüssels, sowie ein Zertifikat für den öffentlichen Schlüssel. Wichtig zu wissen ist, dass sämtliche kryptographischen Operationen innerhalb der Smartcard durchgeführt werden.
Der Kerberosclient sendet das Zertifikat und den Public Key der Smartcard an das KDC. Das KDC prüft das Zertifikat und akzeptiert den öffentlichen Schlüssel. Dann wird der Logon Session Key mit diesem öffentlichen Schlüssel verschlüsselt und mit dem TGT an den Client zurückgeschickt. Dieser kann mit seinem privaten Schlüssel den Logon Session Key entschlüsseln. Von nun an wird mit dem Logon Session Key und dem TGT gearbeitet.

5 Internet Protocol Security (IPSec)

Neu bei Windows 2000 ist auch die Unterstützung von IPSec. IPSec steht für IP Security und ist ein Konzept, mit dem Daten auf der Ebene des IP-Protokolls verschlüsselt werden. Die Verschlüsselung auf dieser Ebene ist für die Anwendungen im Gegensatz zum Beispiel zu SSL vollkommen transparent. Der Clou daran ist: Die Anwendungen wissen nichts von IPSec und müssen dementsprechend auch nicht an die Verwendung von IPSec angepasst werden.

Die Konfiguration von IPSec erfolgt vielmehr über die Gruppenrichtlinien von Windows 2000. Dabei hat der Administrator die Möglichkeit, sehr differenziert festzulegen, was wie verschlüsselt wird. So kann beispielsweise die Kommunikation von Servern mit anderen Systemen über bestimmte Ports vollkommen deaktiviert werden. Welche Ports und Protokolle verschlüsselt werden sollen und mit welchen Systemen überhaupt eine Kommunikation erfolgen darf, kann ebenfalls definiert werden. Für jede zulässige Kommunikation lässt sich wiederum das Sicherheitsniveau definieren.

Die Nutzung von IPSec in Windows 2000 erfordert sehr differenzierte Konzepte, damit man vermeidet, dass mehr Sicherheitslücken oder Zugriffsprobleme entstehen als verhindert werden. Wenn IPSec allerdings richtig genutzt wird, steht es im Bereich der B2B-Kommunikation als effizienter Mechanismus zur Verfügung.

6 Zertifikate

Windows 2000 unterstützt eine sogenannte Infrastruktur öffentlicher Schlüssel. Unter Infrastruktur öffentlicher Schlüssel, auch als PKI (Public Key Infrastructure) abgekürzt, wird ein System digitaler Zertifikate, Zertifizierungsstellen (CAs) und anderer Registrierungsstellen verstanden. Mit Hilfe dieses Systems wird die Gültigkeit der an einer Transaktion beteiligten Parteien anhand der Verschlüsselung mit öffenlichen Schlüsseln überprüft und authentifiziert. Die Standards für die Infrastruktur öffentlicher Schlüssel sind noch in Entwicklung, obwohl diese bereits weitreichend als ein notwendiger Bestandteil des E-Commerces implementiert sind.
Es bestehen mehrere Gründe, die bei der Verwendung von Windows 2000 für den Einsatz einer Infrastruktur öffentlicher Schlüssel in einem Unternehmen sprechen:

- **Umfassende Sicherheit**. Das Verwenden von Smartcards stellt eine sehr sichere Authentifizierungsmethode dar. Vertraulichkeit und Integrität der über das Netzwerk übertragenen Daten können mit Hilfe von IPSec (Internet Protocol Security), die Vertraulichkeit gespeicherter Daten kann durch EFS gewährleistet werden. Ausserdem kann ein Administrator unter Windows 2000 anhand von Sicherheitsberechtigungen die Ausgabe der jeweiligen Zertifikate an Benutzer festlegen.
- **Vereinfachte Verwaltung**. Man kann sich in einem Unternehmen für die Ausgabe von Zertifikaten anstelle von Kennwörtern entscheiden. Zertifikate können jederzeit wiederrufen und Zertifikatssperrlisten veröffentlicht werden. Es stehen auch flexible Verwaltungsmöglichkeiten für Vertrauensbeziehungen zwischen Unternehmen zur Verfügung. Schliesslich wird auch das Zuordnen von Zertifikaten zu Benutzerkonten in Active Directory, oder über IIS (Internet Information Services), unterstützt.
- **Neue Möglichkeiten**. Dateien und Daten können sicher über das öffentliche Netz wie das Internet ausgetauscht werden.
  Weiterhin kann man Nichtverleugnung durch den Einsatz digitaler Signaturen, einem wichtigen Bestandteil des E-Commerces, ermöglichen.

Unter Windows 2000 stehen folgende Features für das Implementieren einer Infrastruktur öffentlicher Schlüssel zur Verfügung:

- **Zertifikate**. Bei einem Zertifikat handelt es sich im Wesentlichen um eine von einer Instanz ausgegebene, digitale Erklärung. Hierbei bürgt eine Instanz für die Identität des Zertifikatinhabers. Über das Zertifikat wird ein öffentlicher Schlüssel und die Identität der Person, des Computers, oder des Dienstes gebunden, zu dem der private Schlüssel gehört. Zertifikate werden in einer Vielzahl von Sicherheitsdiensten und –anwendungen verwendet, die durch den Einsatz öffentlicher Schlüssel Authentifizierung, Datenintegrität und sichere Kommunikation über Netzwerke wie das Internet gewährleisten.

  Das unter Windows 2000 für Zertifikate verwendete Standardformat trägt die Bezeichnung X.509v3. Ein X.509-Zertifikat enthält Informationen über die Person oder Organisation, für die das Zertifikat ausgestellt wurde, sowie Informationen über das Zertifikat und ausserdem optionale Informationen über die ausstellende Zertifizierungsstelle. Die Informationen zur Person oder Organisation können den jeweiligen Namen, den öffentlichen Schlüssel, den Algorithmus für den öffentlichen Schlüssel und eine optionale eindeutige Kennung umfassen. Standarterweiterungen für Zertifikate der Version 3 liefern Informationen zu Schlüsselkennungen, zur Verwendung des Schlüssels, Zertifikatsrichtlinien, weitere Namen und Attribute, Beschränkungen des Zertifizierungspfades und erweiterte Möglichkeiten für die Zertifikatssperrung. (Dies umfasst u.a. Sperrgründe und eine Aufteilung der Zertifikatssperrliste durch Erneuerung der Zertifikatsstelle). Die Anwendung der Zertifikate ist also vergleichbar mit denen von PGP aus der Vorlesung Netzwerksicherheit.

- **Zertifikatsdienste** unter Windows 2000 Server. Unter Windows 2000 wird die Komponente Zertifikatsdienste zum Erstellen und Verwalten von Zertifizierungsstellen (Certification Authorities = CAs) verwendet. Eine CA ist für das Festlegen der Identität einer Person zuständig und bürgt im weiteren Verlauf für die Identität der Zertifikatsinhaber. Weiterhin kann eine CA Zertifikate sperren, wenn diese ungültig werden, und Zertifikatssperrlisten herausgeben, die anschliessend für die Zertifikatsprüfung herangezogen werden. In dem einfachsten Modell einer Infrastruktur öffentlicher Schlüssel ist nur eine einzige CA vorhanden. Tatsächlich verwenden die meisten Unternehmen jedoch eine Reihe von CAs, die in vertrauenswürdigen Gruppen zusammengefasst werden. Diese Gruppen werden als Zertifizierungshierarchien bezeichnet.

- **Richtlinien öffentlicher Schlüssel.** Mit dem Feature Gruppenrichtlinie in Windows 2000 kann man Zertifikate automatisch auf Computer weiterleiten, Zertifikatsvertrauenslisten und allgemein vertrauenswürdige Zertifizierungsstellen einrichten und verwalten.

7 Sicherheitslöcher

Absolute Sicherheit gibt es nicht. Dies ist ein Grundsatz und gilt auch für Windows 2000. Bereits sind verschiedene Dokumente auf dem Internet verfügbar, die Sicherheitslöcher dieses Betriebsystems aufzeigen.
Ist zum Beispiel nur die Standartinstallation gemacht können mit einer sogenannten Null-Sitzung und einem einfachen Tool die Benutzernamen ausfindig gemacht werden. Natürlich können dagegen Einstellungen gemacht werden, aber das muss man zuerst wissen.
Laut Marc Ruef von Computec ist auch das Abfangen von Passwortsequenzen ganz einfach:
„Das L0phtcrack SMB Paket-Abfang-Utility kann nach wie vor die NTLM-Beglaubigungen abfangen und knacken, die zwischen einem NT 4-Client und einem 2000-Server übertragen werden. Auch die Kerberos-Authentifizierung wurde vom Unternehmen aus Redmond so konzipiert, dass die Beglaubigung auf NTLM herabgesetzt werden kann, wenn einer der Kommunikations-Parteien Kerberos nicht unterstützt: Alle Windows NT 4-Rechner machen also indirekt auch die Windows 2000-Systeme unsicher.
Ein Angreifer könnte nun die starke Authentifizierung in einer Windows 2000-Domäne mittels SYN-Flooding auf TCP-Port 88 (Kerberos) am Domänen-Controller unterlaufen, da alle Clients auf die wackelige NT-Beglaubigungsroutine herabgesetzt werden. Das Schnüffeln ist dann nur noch ein Kinderspiel.“

Weitere Sicherheitslöcher und eventuelle Gegenmassnahmen sind auf der Seite von Herrn Ruef aufgelistet (siehe Quellenverzeichnis).

8 Fazit

Fakt ist zunächst, dass Windows 2000 deutlich mehr Sicherheit bietet, als es der Ruf des Systems vermuten lässt. Aus mehreren Dokumenten ist ersichtlich, dass Windows 2000 einen gewaltigen Schritt in Sachen Sicherheit gemacht hat. Im Vergleich mit anderen Betriebssystemen kann Windows 2000 durchaus mithalten.
Zudem ist bei Windows 2000 die Sicherheit leichter zu administrieren. Das gilt insbesondere für die Verteilung von Sicherheitseinstellungen auf Clients, aber auch auf Servern.
Dennoch wird es ohne jeden Zweifel auch für Windows 2000 wieder eine Reihe von Sicherheits-Patches geben, denn absolute Sicherheit gibt es nicht. Dies haben die Attacken auf eine Reihe prominenter Web-Seiten im Februar erneut deutlich gemacht. Es liegt aber auch daran, dass sich Implementierungsfehler nie völlig vermeiden lassen.
Letztendlich kann man aber sagen, wer Windows 2000 richtig administriert, kann ein sehr hohes Sicherheitsniveau erreichen.

9 Quellenverzeichnis

Microsoft WhitePapers auf www.windows.com/windows2000/de
www.windows2000.ch/de

Kerberos www.rvs.uni-hannover.de/arbeiten/studien/sa-lachuette-html/node158.html

Verschiedene Dokumente www.franklin-net.com/WISSEN/N0004.shtml
www.computec.ch/mruef/texte/windows2000.html