Inside Out Attacke
DNS Tunnel

Abstract & Management Summery

Erstellt am 10. Dezember 2003

Von:

Christian Bernet <inside-out@bernet-ch.ch>
René Herrmann <inside-out@gmx.ch>

Diese Arbeit wurde an der Hochschule Rapperswil, Abteilung Informatik im Auftrag der Compass Security durchgeführt.

HSR
HOCHSCHULE FÜR TECHNIK
RAPPERSWIL

http://www.hsr.ch

Abstract

Gemäss verschiedensten Studien werden 80% der Computer Attacken von internen Netzwerken der Unternehmen gestartet. Zum einen geht man davon aus, dass diese Attacken von frustrierten und in IT Belangen kompetenten Angreifern ausgehen - zum anderen werden auch Personen und Computer durch Viren und Trojaner unbewusst zu potenziellen Hackern.

Diese Diplomarbeit lässt die Frage offen, **wie** ein potenzielles bösartiges Programm im internen Netzwerk einer Unternehmung ausgeführt wird. Es interessiert die Frage, **was** das bösartige Programm anstellen könnte. Die Zielsetzung eines Trojaners lässt sich in folgende drei Kategorien gliedern:

- Beschaffung, Veränderung und Zerstörung von Daten
- Versand von Daten ins Internet
- Initiierung einer Remote-Control Verbindung, damit ein Hacker im Internet das korrumpierte System interaktiv kontrollieren kann

Im Fachjargon werden obige Attacken als **"Inside-Out"** bezeichnet, da der Ursprung der Hackeraktivität im internen Netzwerk liegt. Geht es aus Angreifersicht um einen erfolgreichen Aufbau einer Remote-Control Verbindung, bedient man sich der sogenannten **Tunneling-Verfahren**. Diese benutzen ein Protokoll, um den Firewall zu umgehen. Typische Protokolle sind HTTP, Mail oder DNS.

Aus der Sicht von Compass Security sind solche **Inside-Out Tunneling Attacken** eine grosse Bedrohung. Der Fachwelt ist jedoch zuwenig bewusst, welches Potenzial solche Inside-Out Attacken haben. Im Rahmen dieser Arbeit wurde ein Windows Programm und ein Linux DNS-Tunnel Server realisiert. Über DNS-Anfragen baut das Windows Programm einen Tunnel zu dem im Internet stehenden DNS-Tunnel Server auf. Dieser Tunnel wird als Test-Case aufgebaut und lässt sich über eine Webseite steuern.

Damit steht erstmals im Internet ein Test-Tool zur Verfügung, das die Verwundbarkeit gegenüber einer DNS Tunnel Inside-Out Attacke testet. Der DNS Tunnel bietet folgende Funktionalität:

- Übertragung von Dateien vom DNS Tunnel Server zum Client
- Übertragung von Dateien vom Client zum DNS Tunnel Server
- Ausführen von Programmen auf dem Client

Vor der Implementation wurde analysiert, in welche Felder des DNS Protokolls eigene Daten gepackt werden können. Für die Übertragung vom Client zum Server besteht einzig die Möglichkeit, die Daten in dem Feld unterzubringen, das den Hostnamen enthält ("Query Name field"). Für die Datenübertragung vom Server zum Client wird ein Feld genutzt, das normalerweise eine Beschreibung zum DNS-Eintrag enthält ("TXT field").

Um einen maximalen Schutz gegenüber Missbrauch zu gewährleisten, wird bei der Ausführung des Client Tools eine Warnung ausgegeben, die der Benutzer bestätigen muss. Zudem ist die Internet Adresse des DNS Tunnel Servers im Client fest einkompiliert. Es ist eine wichtige Anforderung an diese Diplomarbeit, dass das Produkt als Instrument zur Security Sensibilisierung verwendet werden kann und als echtes Hacker-Tool versagt.

Management Summary

Einleitung

Beim Anschluss eines Netzwerks ans Internet, wird es als selbstverständlich angesehen das Netzwerk gegen unerwünschte Zugriffe aus dem Internet zu sichern. Eine Möglichkeit Arbeitsplätze zu schützen ist zwischen dem Internet und dem internen Netzwerk eine Firewall zu platzieren.

In den wenigsten Fällen wird berücksichtig, dass 80% der Angriffe aus dem internen Netzwerk erfolgen. Einerseits geht man davon aus dass diese Angriffe von frustrierten Mitarbeitern erfolgen. Andererseits werden Benutzer die ein bösartiges Programm einfangen unbewusst zu potentiellen Hackern. Ein simpler „Click“ in einer Dialogbox, die während dem „Surfen“ erscheint oder das Doppelklicken eines Attachments reichen aus, um sämtliche Schutzwirkungen des Firewalls ausser Kraft zu setzen.

Diese Diplomarbeit lässt die Frage offen, wie ein potentiell bösartiges Programm ins interne Netzwerk eingeschleust und ausgeführt wird. Vielmehr interessiert die Frage, was ein bösartiges Programm alles anstellen könnte. Mögliche Ziele eines derartigen Programms lassen sich in drei Kategorien gliedern:

- Beschaffung, Veränderung und Zerstörung von Daten
- Versand von Daten ins Internet
- Initiierung einer Remote-Control Verbindung, damit ein Hacker im Internet das korrumpierte System interaktiv kontrollieren kann

Im Fachjargon werden obige Attacken als “Inside-Out“ Attacken bezeichnet, da der Ursprung des Angriffs im internen Netzwerk liegt. Die attraktivste Kategorie aus der Sicht des Hackers ist der Aufbau einer Remote-Control Verbindung. Wie eine Remote-Control Verbindung aufgebaut werden kann ist von

den Sicherheitsvorkehrungen des Netzwerks abhängig. Als minimale Sicherheitsvorkehrungen wird im Folgenden der Einsatz einer Firewall angenommen.

Eine Firewall bestimmt welche Internetdienste von Rechnern innerhalb des Netzwerks genutzt werden dürfen. Ein Beispiel für einen Internetdienst wäre das World Wide Web, der das „Surfen“ im Internet ermöglicht. Jeder dieser Internetdienste benutzt einen eigenen Kanal, der auf der Firewall freigeschaltet werden muss. Jeder Kanal besitzt zwei Öffnungen: Eine Öffnung ins interne Netzwerk und eine ins Internet. Jede Kanalöffnung besitzt eine Nummer, die sogenannte Portnummer. Der Kanal, der für das „Surfen“ im World Wide Web von einem Rechner im internen Netzwerk angesprochen wird ist der Port 80. Abbildung 1 zeigt die Kommunikation eines internen Rechners über eine Firewall mit einem Webserver.

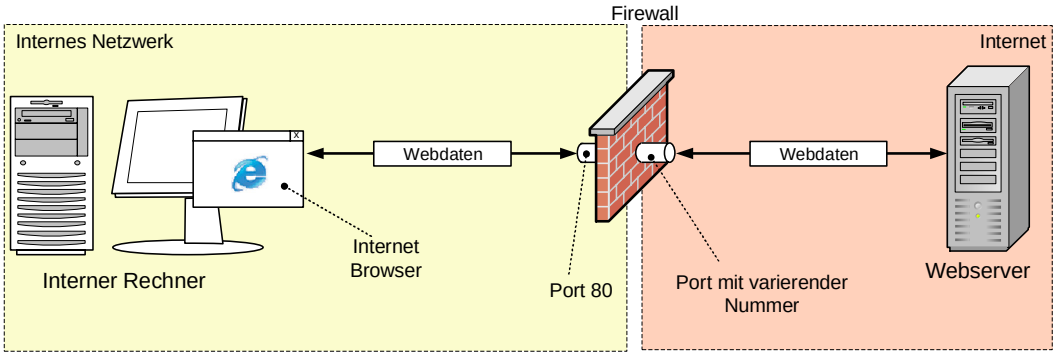

Abbildung 1: Kommunikation über einen offenen Port

Erfolgt von einem internen Rechner eine Anfrage für eine Webseite, wird auf der Firewall für eine bestimmte Zeit ein Port ins Internet geöffnet. Über diesen geöffneten Port gelangt die Antwort eines Webservers zurück zum internen Rechner. Dies gilt auch für Kanäle anderer Internetdienste, die für interne Rechner freigeschaltet sind.

Über alle Kanäle, die auf der Firewall für interne Rechner freigeschaltet sind kann ein bösartiges Programm eine Remote-Control Verbindung zu einem Rechner ins Internet herstellen. Die Firewall kontrolliert beispielsweise nicht, ob auf Port 80 tatsächlich eine Anfrage für eine Webseite eingegangen ist und ob die Antwort aus dem Internet auch wirklich eine Webseite darstellt.

Angenommen es handelt sich beim internen Netzwerk, um ein besser gesichertes Netzwerk. Beispielsweise ist es so eingerichtet, dass tatsächlich nur Daten ausgetauscht werden dürfen, die dem erlaubten Internetdienst entsprechen. Dies wird dadurch möglich, dass die Daten eines Dienstes in dem für diesen Dienst vorgesehenen Format vorliegen. Für jeden Dienst ist ein so genanntes Protokoll definiert. In diesem Protokoll steht wann während des Kommunikationsvorgangs, welche Daten in welchem Format übertragen werden. Für das „Surfen“ wird beispielsweise das Hypertext Transfer Protokoll (HTTP) genutzt.

Versucht ein bösartiges Programm in einem auf diese Weise gesicherten Netzwerk eine Verbindung nach Draussen herzustellen wird sein Datenverkehr nicht durch die Firewall gelassen, wenn sie nicht dem Protokoll entsprechen. Das Programm besitzt allerdings die Möglichkeit seine Daten in das erlaubte Format zu kapseln. Werden Daten in ein Protokoll gekapselt spricht man von „Tunneling“.

Diese Arbeit befasst sich insbesondere mit der Funktionsweise des DNS-Tunnels (DNS steht für „Domain Name Server“). Dieser Dienst wird immer dann genutzt, wenn ein Rechner im Internet mit einem anderen Rechner kommunizieren will, aber nur den Rechnernamen kennt. Beispielsweise fragt ein Rechner den Domain Name Server (DNS) nach der IP-Adresse des Rechners „www.hsr.ch“. Die Antwort des DNS enthält die IP-Adresse „152.96.37.80“. Bei einer solchen Namensauflösung können verschiedene Rechner beteiligt sein. Abbildung 2 zeigt ein vereinfachtes Beispiel für den Ablauf einer DNS-Anfrage.

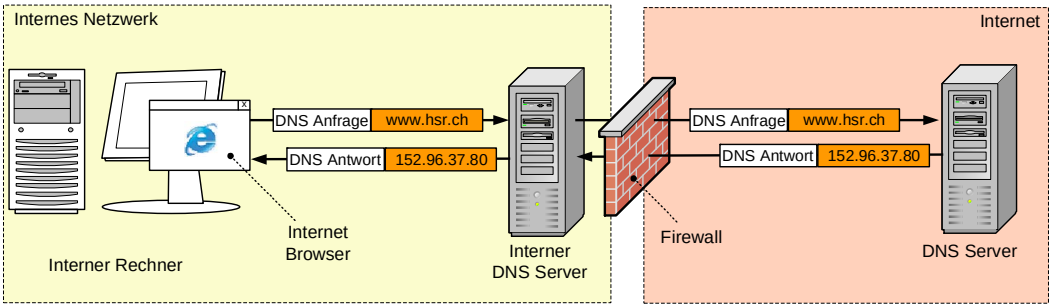

Abbildung 2: Ablauf einer normalen DNS-Anfrage

Bei der Benutzung eines DNS-Tunnels werden in den DNS-Nachrichten zusätzlichen Daten untergebracht. In der Abbildung 3 werden der Anfrage des Rechnernamens die Daten „xyz“ vorangestellt. Die Antwort enthält anstatt der IP-Adresse die Daten „abc“.

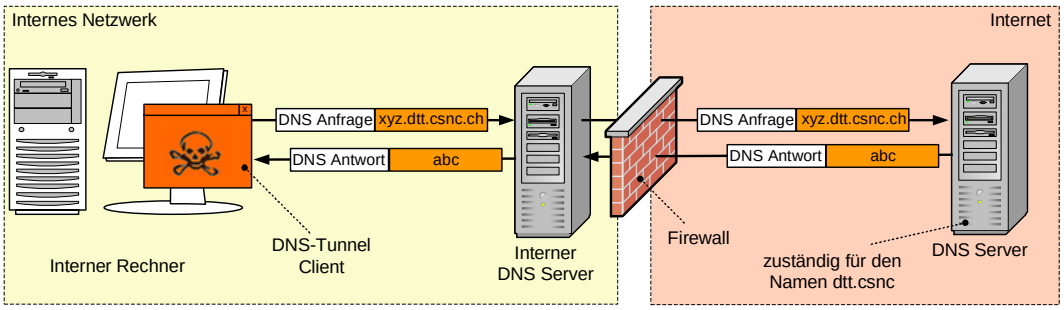

Abbildung 3: Ablauf einer DNS-Anfrage des Tunnels

Ausgangslage

Aus der Sicht von Compass Security sind solche Inside-Out Tunneling Attacken eine grosse Bedrohung. Der Fachwelt ist jedoch zu wenig bewusst, welches Potenzial solche Inside-Out Attacken haben.
Das Ziel dieser Diplomarbeit ist die Fachwelt für die Inside-Out Problematik zu sensibilisieren. Um dieses Ziel zu erreichen wurde eine Testsuite eingerichtet. Die Testsuite verwendet für die Kommunikation einen DNS-Tunnel.

Es gibt bereits einen DNS Tunnel mit dem Namen NSTX [1]. NSTX bietet einen DNS-Tunnel zur reinen Datenübertragung an.
Im ersten Teil der Arbeit wurde eine Publikation zum Thema Inside-Out Attacken erstellt. Diese Publikation informiert über die verschiedenen Attacken Arten, das Gefahrenpotenzial und mögliche Schutzmassnahmen. Im zweiten Teil wurde eine DNS-Tunnel Testsuite implementiert, die es jedem ermöglicht das eigene Netzwerk auf diese Inside Out Attacke zu testen. Die Testsuite bietet folgende Funktionalitäten an:

- Übertragung von Dateien vom DNS Tunnel Server zum Client
- Übertragung von Dateien vom Client zum DNS Tunnel Server
- Ausführen von Programmen auf dem Client

Vorgehen

In einem ersten Teil der Arbeit wurden verschiedene Varianten aufgezeigt, wie ein bösartiges Programm die vorhandenen Sicherheitsmassnahmen umgehen kann. Es stellte sich heraus, dass es entweder die Möglichkeit gibt eine simple direkte Verbindung über einen Port zu erzeugen, falls der Rechner einen direkten Internet Zugang besitzt oder dass die Daten innerhalb eines erlaubten Protokolls zu kapseln. Die verschiedenen Varianten wurden ausführlich dokumentiert und in Form einer Publikation für Compass Security beschrieben.
Im zweiten Teil der Arbeit wurden die Möglichkeiten für die Umsetzung eines Tunnels studiert, der das DNS-Protokoll benutzt. Dazu wurde eine Studie durchgeführt, die zeigte, in welcher Weise Daten in DNS-Anfragen und DNS-Antworten verpackt werden können. Eine Analyse des DNS-Standards (RFC-1035) [3] zeigte, dass es in einer DNS-Nachricht verschiedene Felder

gibt, die Daten enthalten. Zum Beispiel enthält eine DNS-Anfrage zwecks Namensauflösung ein Feld für den Hostnamen, der in eine IP-Adresse umgewandelt wird. Allerdings kann ein solches Feld nur bestimmte Zeichen enthalten. Diese Fragen mussten also zuerst geklärt werden. Getestet wurden die Erkenntnisse anschliessend mittels verschiedener Prototypen. Das Endprodukt besteht aus einem Windows-Programm (Client), der möglichst einfach auf einem Rechner in einem Netzwerk installiert werden kann und einem Linux-Programm (Server) der die Anfragen des Clients beantwortet.
Um die Bedrohung zu demonstrieren die von einem bösartigen Programm ausgeht, der die Fähigkeit besitzt einen solchen Tunnel einzurichten, wurde der Server mit einem Webinterface ausgestattet, der es erlaubt Befehle an den Client zu senden, die der Client ausführt und den Output an den Server zurückschickt.

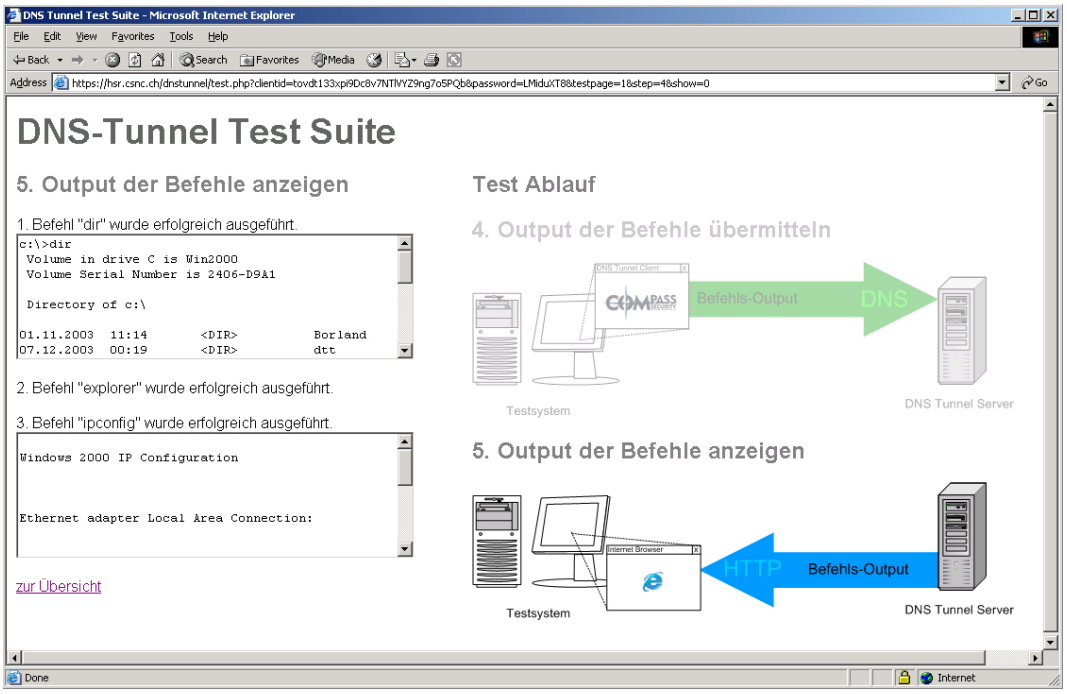

Abbildung 4: DNS Testsuite

Um einen maximalen Schutz gegenüber Missbrauch zu gewährleisten, wird bei der Ausführung des Client Tools eine Warnung ausgegeben, die der Benutzer bestätigen muss. Zudem ist die Internet Adresse des DNS Tunnel Servers im Client fest einkompiliert. Es ist eine wichtige Anforderung an diese Diplomarbeit, dass das Produkt als Instrument zur Security Sensibilisierung verwendet werden kann und als echtes Hacker-Tool versagt.

Ausblick

Gelingt es einem Angreifer ein bösartiges Programm auf einen Rechner einzuschleusen und es existiert irgendeine Kommunikationmöglichkeit vom internen Netzwerk zum Internet, so wird es immer möglich sein, eine Remote-Control Verbindung aufzubauen. Die einzige Möglichkeit sich vor dieser Art von Attacke vollständig zu schützen ist das interne Netzwerk und das Internet vollständig voneinander abzutrennen.
Der Client der im Rahmen dieser Arbeit entwickelt wurde ist bereits sehr einfach in seiner Handhabung. Von einer applikationsbasierten Firewall wird er jedoch entdeckt. Eine applikationsbasierte Firewall entlarvt Programme, die sich mit dem Internet verbinden wollen. Eine denkbare Erweiterung wäre, den Client so auszubauen, dass er eine applikationsbasierte Firewall täuscht, so wie dies im Artikel „Firehole“ [2] von Robin Keir beschrieben wird. Dabei wird das bösartige Programm in den selben Speicherbereich kopiert, wie ein anderes Programm (z.B. der Internet Explorer), das berechtigt ist aufs Internet zuzugreifen.

Literaturverzeichnis

[1] Gergely Nagy, Tamas Szerb. Nstx - tunneling network-packets over dns. Website: `http://savannah.nongnu.org/projects/nstx/`.

[2] Robin Keir. How to bypass your personal firewall outbound detection. Website: `http://keir.net/firehole.html`.

[3] P. Mockapetris. Domain names - implementation and specification. Website: `http://www.ietf.org/rfc/rfc1035.txt`.