

Sicherheit in Netzwerken (1)

Accounts und Paßwörter

Eine starke Paßwortsicherheit ist das billigste und mächtigste Mittel, um sich gegen unerlaubte Systemnutzung zu verteidigen. Angreifer versuchen, unautorisierten Zugang zu Computersystemen zu erlangen, indem sie die Paßwörter von legitimen Nutzern raten oder „knacken“. Wenn ein Angreifer Erfolg hatte, hat er die Möglichkeit, sich im System nach anderen Sicherheitslöchern umzuschauen, Superuserrechte zu erlangen oder im Namen des ursprünglichen Account-Inhabers Handlungen durchzuführen. Der beste Weg, sich vor unautorisiertem Zugriff zu schützen, ist der Schutz vor solchen Angreifern. Daraus ergeben sich einige Punkte, die im folgenden Text näher erläutert werden.

Jede Person, die ein UNIX-System benutzt, sollte einen *Account* haben. Ein Account besteht aus zwei Teilen: einem *Username* und einem *Paßwort*. Der Username, auch Accountname oder Login genannt, ist ein *Identifikator*. Er teilt dem System mit, wer es gerade benutzt. Das Paßwort ist ein *Authentifizierer*. Es erlaubt dem System zu überprüfen, ob der derzeitige Nutzer auch der ist, der er vorgibt zu sein.

Eine Person kann mehr als einen UNIX-Account haben, jeder mit seinem eigenen Username. Usernamen sind einheitliche Namen, die zwischen 1 und 8 Zeichen lang sein können. Ein Username identifiziert den Nutzer genauso, wie er sich mit seinem Vornamen bei seinen Bekannten oder Freunden z.B. am Telefon identifiziert. Wenn kein Alias definiert wurde, wird eine E-Mail an diesen Nutzer, in der Regel an seinen Usernamen adressiert.

UNIX benutzt die Datei */etc/passwd*, um die Informationen zu jedem Nutzer zu speichern. Diese Datei enthält den Username, das verschlüsselte Paßwort, die User Identification Number (UID), die Group Identification Number (GID), den Realnamen, das HOME-Verzeichnis und seine Shell. Diese sieben Felder sind durch einen Doppelpunkt voneinander getrennt.

```
alex:fi3sED95ibqR6:100:10:Alexander Geschonneck:
/home/alex:/bin/tcsh
```

Das Paßwort ist in einer speziellen Form gespeichert. Bei einigen UNIX-Systemen kann das Paßwort auch in einer gesondert gesicherten Datei, der Shadow-Paßwortdatei, gespeichert sein.

Was versteht man unter Authentifizierung?

Nachdem der Nutzer dem UNIX-System mitgeteilt hat, wer er ist, überprüft UNIX dies mit einem Prozeß, der Authentifizierung genannt wird. Die meisten Leute denken bei Authentifizierung an Paßwörter. Paßwörter werden zwar häufig zur Authentifizierung benutzt, es

gibt jedoch eine Vielzahl weiterer Authentifizierungsmechanismen. Diese lassen sich im allgemeinen mit Hilfe der folgenden Kriterien klassifizieren:

<i>Etwas, das der Nutzer weiß.</i>	Das ist das traditionelle Paßwortsystem.
<i>Etwas, das der Nutzer hat.</i>	Dazu gehören verschiedene Mechanismen, z.B. Frage-Antwort-Listen, Einmal-Paßwörter, Keycards, usw.
<i>Etwas, das zum Nutzer gehört.</i>	Das ist das Einsatzgebiet der Biometrik. Zu dieser Gruppe gehören Fingerabdrücke, Merkmale der Netzhaut, Stimmenanalyse usw.

Einige Systeme kombinieren mehrere Ansätze. Eine Keycard, deren Benutzer eine persönliche Identifikationsnummer (PIN) eingeben muß, kombiniert zum Beispiel etwas, das man hat (die Keycard) und etwas, das man weiß (die PIN). Theoretisch ist es vorteilhaft, mindestens zwei Verfahren zu kombinieren. Das, was der Nutzer hat, kann gewöhnlich Diebstahl zum Opfer fallen; das, was der Nutzer weiß, ist bei der Übertragung im Internet durch „Paketschnüffler“ in Gefahr. Es ist jedoch selten, daß ein Angreifer beides gleichzeitig erhält.

Zur Zeit sind viele biometrische Systeme im Einsatz oder in der Entwicklung. Sie überprüfen so unterschiedliche persönliche Merkmale wie Stimme, Finger- oder Handabdrücke, Netzhautmerkmale, Unterschrift oder die Art und Weise, wie der Nutzer tippt. Biometrische Systeme genießen ein enormes Interesse, da sie das Problem umgehen, daß etwas gestohlen oder aufgedeckt werden kann. Selbst das Horrorszenario eines abgehackten Daumens wird berücksichtigt: bei den meisten Fingerabdruck-Scannern muß ein Puls vorhanden sein. Leider sind die meisten biometrischen Systeme für normale Internet-Anwendungen nicht praktikabel.

Der Anmeldevorgang

Heutzutage richten sich die meisten UNIX-Systeme nach der ersten Authentifizierungsstrategie - etwas, das der Nutzer weiß. Das Paßwort ist ein Geheimnis, daß der Nutzer mit dem Computer teilt. Wenn sich ein Nutzer bei einem UNIX-System anmeldet, muß er ein Paßwort eingeben, damit das System überprüfen kann, ob der Nutzer der ist, der er vorgibt zu sein. Das Paßwort wird nicht angezeigt, während der Nutzer es eingibt. Dies gibt dem Nutzer Schutz, wenn er von ei-

nem potentiellen Angreifer beim Anmelden beobachtet wird.

Paßwörter sind die erste Verteidigungslinie eines UNIX-Systems, um sich gegen Angreifer zu schützen. Nebenbei ist es natürlich möglich, über ein Netzwerk in ein System einzubrechen, ohne sich vorher anzumelden. Statistisch gesehen beruhen aber 80 % aller Angriffe auf Unzulänglichkeiten in der Paßwortwahl.

Wenn UNIX nach dem Paßwort des Nutzers fragt, braucht es einige Mechanismen, um zu überprüfen, ob das eingegebene Paßwort auch korrekt ist. Einige frühe Computersysteme speicherten die Paßwörter ihrer Nutzer unverschlüsselt in einer Datei, die nur vom Superuser oder von Systemroutinen bearbeitet werden konnte. Durch Systemfehler war diese aber auch für alle anderen Nutzer sichtbar, z.B. wenn der Superuser diese Datei mit einem Editor bearbeitete und somit während der Bearbeitung eine für jedermann lesbare temporäre Datei anlegte. Das Problem bei diesem Vorgehen war das unverschlüsselte Speichern der Paßwörter!

Moderne UNIX-Systeme hingegen speichern in einer Datei einen Wert, der durch das Verschlüsseln eines Blockes von Nullen in einer nichtrekursiven *crypt*-Funktion entstanden ist. Hierbei ist das Paßwort der Schlüssel. Das Ergebnis dieser Rechenoperation wird normalerweise in der Datei */etc/passwd* gespeichert. Wenn sich nun ein Nutzer in einem UNIX-System anmeldet, entschlüsselt das Programm */bin/login* nicht etwa den gespeicherten verschlüsselten Wert in */etc/passwd*. */bin/login* benutzt hingegen das eingegebene Paßwort, um eine vorgegebene Anzahl von Nullen zu verschlüsseln und vergleicht diesen so errechneten Wert mit dem Eintrag des zweiten Feldes der */etc/passwd*. Wenn beide Werte übereinstimmen, hat der Nutzer Zugang zum System.

Der *crypt*-Algorithmus ist bis heute einigermaßen resistent gegenüber Attacken. Er beruht auf dem Data Encryption Standard (DES) des National Institute of Standards and Technology (NIST). Bei normalen Operationen nutzt DES einen 56-bit-Schlüssel (z.B. acht 7-bit-Zeichen), um einen Originaltext zu verschlüsseln, der 64 bit lang ist. Der entstandene 64-bit-Block verschlüsselten Textes kann nicht ohne weiteres (selbst eine „Brute Force Attack“ hilft nicht unbedingt weiter) entschlüsselt werden, wenn der originale 56-bit-Schlüssel nicht bekannt ist.

Die UNIX-*crypt*-Funktion nimmt das Paßwort des Nutzers als Schlüssel und verschlüsselt mit ihm einen 64-bit-Block mit Nullen. Der daraus resultierende 64-bit-Block verschlüsselten Textes wird nun erneut mit dem Nutzerpaßwort als Schlüssel verschlüsselt. Das Ganze wiederholt sich 25mal. Die nun entstandenen 64 bit werden in eine Zeichenkette mit 11 druckbaren Zeichen verwandelt und im zweiten Feld der */etc/passwd* hinterlegt.

Die Quellcodes des *crypt*-Algorithmus sind frei verfügbar, und bisher wurde keine Möglichkeit gefunden bzw. veröffentlicht, das verschlüsselte Paßwort wieder in das Original-Paßwort zu transferieren. Der einzige Weg, den UNIX-Paßwortmechanismus zu unterlaufen, ist die Möglichkeit, das Paßwort zu erraten, es mit dem *crypt*-Algorithmus zu verschlüsseln und das Ergebnis mit dem Eintrag von */etc/passwd* zu vergleichen. Der Erfolg von solchen Paßwortattacken hängt natürlich von umfangreichen Wörterbuchdateien und leistungsfähiger Hardware ab.

Natürlich muß verhindert werden, daß die Paßwort-Einträge von zwei Nutzern in der */etc/passwd*, die zufällig das gleiche Paßwort benutzen, nicht identisch sind. Auf diesem Wege könnte einer dieser beiden Nutzer den Account des anderen benutzen, sobald er herausfindet, daß sie beide das gleiche Paßwort haben. Aus diesem Grunde wurde das „Salt of the Day“ eingeführt. Dieses DES Salt ist eine 12-bit-Zahl zwischen 0 und 4095. Jedes dieser verschiedenen Salts ermöglicht 4096 Varianten der Paßwortverschlüsselung. Wenn ein Nutzer sein Paßwort ändert, wählt das Programm */bin/passwd* ein Salt aus, das von der aktuellen Uhrzeit abhängig ist. Das Salt wird in eine zweistellige Zeichenkette konvertiert und mit dem verschlüsselten Paßwort in der */etc/passwd* gespeichert. Dies ist notwendig, weil beim Anmelden das gleiche „Salt of the Day“ zum Verschlüsseln benutzt wird. UNIX speichert das betreffende „Salt of the Day“ als die ersten beiden Zeichen des Textes im zweiten Feld der */etc/passwd*. So werden zwei gleiche Paßwörter als verschiedene verschlüsselte Texte gespeichert, basierend auf der genauen Tageszeit. Wenn nun ein Angreifer ein Paßwortrauteprogramm ansetzt, muß er jeden Versuch, das vermutete Paßwort zu entschlüsseln, 4096 wiederholen. Hier wird nur der Angreifer schnell erfolgreich sein, der über die entsprechende CPU-Leistung verfügt.

Die richtige Wahl

Einige UNIX-Systeme verhindern, daß der Nutzer sich ein unsicheres Paßwort aussucht. Entweder findet eine Überprüfung der Anzahl der Zeichen statt, oder es wird überprüft, wieviel Sonderzeichen, Großbuchstaben und Zahlen enthalten sind (z.B. *npasswd*). Wirkungsvoll sind auch Programme, die simultan eine Datenbank mit schlechten oder bereits geknackten Paßwörtern abfragen (z.B. *npasswd*, *cracklib*). Folgende Paßwörter sollten niemals (*niemals!*) benutzt werden:

1. der eigene Name/Accountname
2. der Name des Lebensgefährten oder Ehegatten
3. der Name der Eltern
4. der Name eines Haustieres
5. der Name des eigenen Kindes
6. der Name eines Freundes oder Kollegen

7. der Name eines beliebigen Künstlers oder eines Politikers
8. der Name des Chefs
9. überhaupt Namen
10. der Name des Betriebssystems
11. der Hostname des Computers
12. Telefonnummern
13. Geburtstage
14. Usernamen des benutzten Computers
15. ein Wort aus einem Wörterbuch (!)
16. Straßen oder Städtenamen
17. einfache Zeichenfolgen wie abcd, 12345, qwertz (auch nicht in Großbuchstaben)
18. alle obengenannten Varianten rückwärts geschrieben oder nur von einer Zahl gefolgt

Gute Paßwörter sollten 8 Zeichen lang sein und aus Groß- *und* Kleinbuchstaben bestehen. Außerdem sollten Sonderzeichen enthalten sein. Das Paßwort sollte so leicht zu merken sein, daß es nicht aufgeschrieben werden muß. Das Eintippen sollte möglichst schnell gehen, so daß niemand den Vorgang beobachten und wiederholen kann.

Einige gute Beispiele sind:

NE14TenS (*anyone for tennis?*)
AuaEGC (*all UNIX-admins eat green cheese*)
A15iZfm (*Am Fünfzehnten ist Zahltag für mich*)
+ek'Hx93
aRon4Q
usw.

Die obengenannten Paßwörter sind jetzt schlechte Paßwörter, weil sie hier veröffentlicht wurden!

Leider ist Vertrauen häufig symmetrisch. Also sollten für verschiedene Systeme auch verschiedene Paßwörter benutzt werden.

Selbstverständlich gelten die Richtlinien für die Wahl von richtigen Paßwörtern auch für Netzwerksoftware, die besonders im DOS/Windows-Bereich Anwendung findet. Bei Banyan VINES sorgt der VINES Security Service dafür, daß die Paßwörter eine ordnungsgemäße Länge haben. Eine Lebensdauer des Paßwortes kann dort standardmäßig auch eingestellt werden.

Alle anmeldenden VINES-Arbeitsplätze verwenden eine modifizierte Nummernfolge der Adresse des VINES-Servers, der als erster auf die Anfrage nach einer eigenen Adresse geantwortet hat. Mit dieser Adresse meldet sich die VINES-Station via Street-Talk-Dienst an. Intern findet dabei ein Logout statt, bei dem etwaige alte Verbindungsinformationen gelöscht werden. Danach erhält die Station vom Server eine Sitzungs-ID. Nun fragt der Rechner Namen und Paßwort des Nutzers ab, verschlüsselt dessen Angaben mit der Sitzungs-ID, löscht das Paßwort aus dem Arbeitsspeicher der Station und verschlüsselt das Resultat noch einmal mit der Stationsadresse, ehe das Ganze zum Server geschickt wird. Dieser berechnet seinerseits den Paßwortschlüssel und vergleicht ihn mit der ankommenden verschlüsselten Anfrage. Ist der Wert identisch, wird das Netz freigeschaltet.

Ein schlechtes Paßwort ist ein Paßwort, das leicht zu raten ist.

Der Angreifer braucht nur *einen* Erfolg, um das gesamte System zu kompromittieren!

Alexander Geschonneck
geschonneck@rz.hu-berlin.de
<http://hppool0.rz.hu-berlin.de/~h0271cbj/index.html>