Abnahme von Eisfair-Paketen

Diskussionsvorschlag:
Ansgar (Ansgar.Puester@T-Online.de)

Version: 1.0
Datum: 25.5.2003

Grundlagen

Eisfair-Pakete werden in der Regel als User eis, d.h. mit einer root-Berechtigung (UID=0) installiert.

Dies ist notwendig, um alle Operationen (Anlegen von Verzeichnisse, Erzeugen von Dateien und Links, aber auch Anlegen von Usern und Gruppen) durchführen zu können.

Die hohe Berechtigung bietet alle Möglichkeiten, enthält aber auch erhebliche Gefahren.

Dies wurde unter anderem am Beispiel des wget4web Paketes deutlich, welches das Executable /usr/bin/wget mit einer inkompatiblen Version überschrieb. Ergebnis war, dass ab dann alle Installationsversuche fehlschlugen. Eine Sicherheitskopie der Originalversion von wget wurde durch das Paket nicht angelegt. Einige User mussten über in der Newsgroup bereitgestellte Hilfslösungen wget restaurieren bzw. installierten ihre Eisserver vollständig neu.

Es lassen sich nahezu beliebige Szenarien skizzieren, die ähnlich fatale Folgen hätten. Dumme Fehler (Löschen /etc/passwd) wären dabei noch relativ offensichtlich.

Böse Folgen für die Akzeptanz von Eisfair hätten z.B. auch Viren, Würmer, Adware-Softwaren oder Backdoor-Programme, die mit einem Eisfair Paket auf einen Eisserver gelangen könnten.

Grundsätzlich ist jeder Administrator eines Eisservers für das verantwortlich, was er mit / auf dem Eisserver tut.
D.h. er und allein er trägt die volle Verantwortung für etwaige Folgen seines Tuns.

Wir sollten aber Ansätze diskutieren, um

- Die „offiziellen“ Pakete möglichst sicher zu gestalten.
- Paketentwicklern Hilfestellungen zu geben.
- Den Eisfair-User (hier eigentlich Administratoren) ihre Verantwortung deutlich zu machen.

Organisatorische Lösungsansätze

Softwareentwicklungsprozess

In kommerziellen Softwareentwicklungsprozessen werden unter Einhaltung der dort definierten Phasen (development, test and integration, acceptance test, production) und durch Nutzung z.B. des Vier-Augen-Prinzips (Tester ungleich Entwickler) in der Regeln die gröbsten Fehler gefunden. Ausnahmen bestätigen allerdings diese Regel ;-)

Für das Eisfair-Team könnte ein Lösungsansatz sein, dass

- Jeder Paketentwickler einen „Paten“ bekommt. Dieser prüft das Paket bevor es im Development Bereich bereit gestellt wird.
- Nach einer noch festzulegenden Zeit im Development Bereich (oder einer noch festzulegenden Zahl von Downloads; wie war das noch mit einem Download-Counter?) wird das Paket in den offiziellen / stabilen Bereich verschoben.

Wir sollten uns auch nicht „schämen“ Hot-Fixes ,z.B. bei Sicherheits-lücken, herauszugeben (Meine Meinung: Lieber ein Hot-Fix als ein „Sch.... wieso ist mir das nicht früher aufgefallen?“).

Für die Pakete von Entwicklern außerhalb des Eisfair-Teams sollte es die Möglichkeit geben das Paket durch Mitglieder des Teams einer Art Prüfung (Abnahme) zu unterziehen.
Hier sollten die Erfahrungen aus dem Fli4l-Projekt (Rolle der Opt-COPs) entsprechend übertragen werden.
Ob sich ggf. Freiwillige für die Rolle des Package-COPs finden werden, die auch über die notwendige „Frei“-Zeit verfügen, wird sich zeigen.

Dokumentation

Die vorhandene Entwicklerdokumentation sollte um ein entsprechendes Kapitel zu Sicherheit- und zu Stabilitätsansprüchen erweitert werden.

Eine Checkliste, die sich ein Paketentwickler vor dem finalen Punkt „Breitstellen des Paketes im Netz“ Schritt für Schritt zu Gemüte führen kann könnte ebenfalls die Qualität verbessern.

Warnungshinweise

Wird in einem offiziellen oder einem fremden Paket ein schwerwiegendes Problem entdeckt, so würde eine Liste mit Warnhinweisen auf www.eisfair.org zumindest verhindern können, dass sich weitere User Probleme einfangen.
Das Paket wget4web wäre z.B. ein Kandidat für eine solche Liste gewesen.

Technische Lösungsansätze

Absichern des Basissystems

Die technische Absicherung des Basissystems gegen fehlerhafte Pakete ist eine recht umfangreiche, andauernde Aufgabe. Es müssten Informationen über zwingend notwendige Files und Directories, deren Rechte und im Grunde auch deren notwendige Minimal-Inhalte (z.B. /etc/passwd) abgelegt werden. Das käme einer Sicherheitskopie gleich. Diese Sicherheitskopie wäre im Übrigen auch nur schwerlich gegen mutwillige Modifikationen zu schützen.
Ein solcher Ansatz könnte zumindest in soweit weiterverfolgt werden, dass die Installationsroutine für Pakete z.B. prüfen könnte wenn eine Art TOP 100-Liste von Files bei Installation eines Paketes überschrieben wird. So sollte z.B. das Überschreiben von /etc/passwd zumindest eine dicke Warnung erzeugen.

Überprüfung der tar.gz-Files

Es wäre mit wenig Aufwand möglich ein kleines Skript oder Programm zum Checken des tar.gz Files der Pakete und des zugehörigen tar.gz.info Files zu schreiben, welches die Existenz der obligaten Dateien (var/install/package/<paket>, tmp/preinstall.sh etc. etc.) überprüft. Zusätzlich könnte der Inhalt des tar.gz.info Files und der Kontext zwischen tar.gz Files und tar.gz.info File geprüft werden.

Stellt man dieses Checkprogramm den Entwicklern zur Verfügung, könnten die Qualität der Installationspakete auf jeden Fall vergrößert werden.

Impaktanalyse mit icheckdiff

Für die Entwicklung der Pakete Squid und Inet benutze ich ein Shell-Skript mit dem Namen icheckdiff.
Es ist auf meine Homepage unter Eisfair-Tools verfügbar.

Mit icheckdiff verfolge ich zwei Ziele:

- Im Laufe des Entwicklungsprozesses können Differenzen zwischen zwei unterschiedlichen Paketversionen auf Basis des fertigen tar.gz-Files, also des Endproduktes, aufgelistet werden.
  Damit ist es möglich z.B. zu erkennen, dass wirklich nur die eine Datei, die ich ändern wollte im neuen Installationspaket steckt. Überflüssige Dateien (wie die beliebten Tilde-Dateien bzw. .swp-Dateien), die sich in einigen Installationspaketen tummeln fallen sofort auf.
  Darüber hinaus überprüft icheckdiff die Existenz einige obligaten Files und warnt bei einigen als curious festgelegten Dateirechten (z.B. Write Rechte bei Files, die root gehören).
- Wird icheckdiff mit nur einem Argument, d.h. einem tar.gz File aufgerufen, so wird die Analyse zwischen Installationspaket und aktuellem Filesystem durchgeführt.

Dies kann direkt auf dem Eisserver zu drei Zeiten erfolgen.

Vor der Installation des Paketes:
Hier ist sofort zu erkennen, wenn eine existierende Datei bei der Installation überschrieben würde. Beim wget Beispiel wäre zumindest gemeldet worden, dass /usr/bin/wget Kandidat für das Überschreiben gewesen wäre.

Nach Installation des Paketes:
Hier kann z.B. geprüft werden, dass die beim Installationsvorgang notwendigen Änderungen an Rechten (chown, chmod) durchgeführt wurden.

Nach der Deinstallation eines Paketes:
Hier kann geprüft werden, ob alle Dateien bei der Deinstallation wirklich gelöscht wurden bzw. ob es (bewusst) zurückgelassene Dateien gibt.

Grosses Manko von icheckdiff ist, dass nur statische, d.h. im tar.gz File wirklich vorhandene Dateien und Verzeichnisse geprüft werden. Eine dynamisch beim Installieren bzw. beim Konfigurieren des Paketes erzeugte Datei kann so nicht geprüft werden.

In Summe hat mir icheckdiff schon gute Dienste geleistet. Es wäre m.E. sinnvoll über ein solches Tool weiter nachzudenken.