mapWOC-Handbuch

Release 1.3.x

25. 09. 2012

Inhaltsverzeichnis

KAPITEL 1

Einführung

Dieses Handbuch beschreibt die Nutzung und technische Hintergründe von mapWOC Version 1.3.x.

1.1 Gefahren auf Webseiten

Webseiten werden zunehmend als Einfallstor für die Infektion der Rechner ihrer Besucher genutzt. Dabei haben die Betreiber ihren Webseiten nicht etwa selbst missbräuchliche Inhalte hinzugefügt. Vielmehr werden sie selbst zu Opfern von Angriffen, bei denen die Inhalte ihrer Seiten verfälscht wurden. Meist handelt es sich nur um ein kleines Iframe-Element, das in die Datenbank des Betreibers eingeschleust wurde und während der Generierung neuer Seiten dem Inhalt unbemerkt hinzugefügt wird.

Während der Darstellung solcher Seiten wird im Browser noch Inhalt von einem zweiten (i.d.R. nicht vertrauenswürdigen) Server nachgeladen. Dieser Inhalt ist dann böswillig und infiziert über Schwachstellen im Webbrowser den Rechner. Der vollständige Vorgang wird auch als Drive-By-Download bezeichnet.

1.2 Was ist mapWOC?

mapWOC dient zur automatisierten Überprüfung der Integrität von Webseiten und der Erkennung böswillig verfälschter Inhalte.

mapWOC steht für massive automated passive Web Observation Center:

- massive: umfassende virtuelle und native Browsersysteme, verwendbar als Einzel- oder hoch skalierbare Cluster-Lösung (bis zu 1 Mio URLs pro Tag pro Knoten)
- automated: automatisiertes Ansurfen von definierten eigenen URL-Listen, analysieren des Netzwerkverkehrs nach Schadsoftware
- passive: Verweilen für definierte Zeit auf jeder URL (Warten auf Angriff)

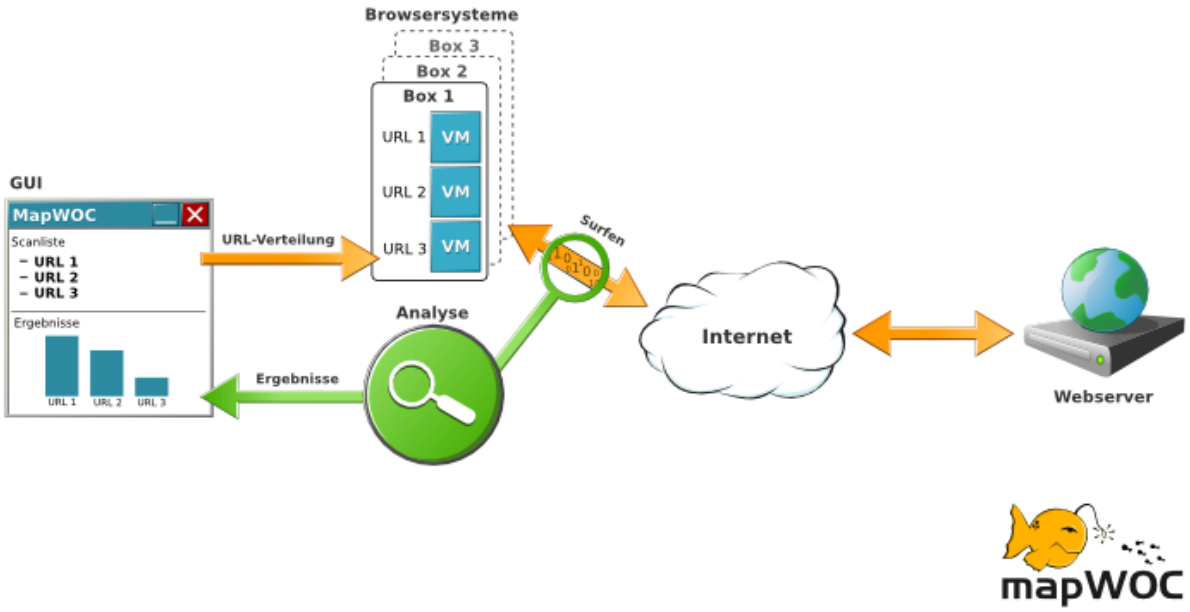
mapWOC wird vom Bundesamt für Sicherheit in der Informationstechnik (BSI) unterstützt.

1.3 Komponenten von mapWOC

mapWOC nutzt u.a. folgende Freie Software Komponenten:

- Debian Squeeze als Hostsystem
- KVM zur Virtualisierung
- ClamAV zur Untersuchung der Schadsoftware

1.4 Funktionsweise von mapWOC



1.5 mapWOC im Cluster-Betrieb

mapWOC kann in einem Rechnerverbund (Cluster) betrieben werden, d.h. mapWOC kann auf mehr als einem Rechner laufen. Die Motivation für das Bilden eines Clusters ist die gesteigerte Performanz, also der Besuch und die Analyse von mehr URLs pro Zeit.

Ein Cluster besteht aus genau einem **Master** und beliebig vielen **Knoten**.

Der **Master** hält die Resultat-Datenbank und die Benutzeroberfläche; über letztere ist eine Interaktion mit den Knoten möglich.

Ein **Knoten** steuert die gewünschten Honey-Clients, analysiert den aufgezeichneten Netzwerkverkehr und überprüft gefundene Dateien auf Viren. Die Ergebnisse werden durch den Master über eine gesicherte Verbindung abgeholt und auf dem Knoten entfernt. Ein Knoten bietet keine Benutzeroberfläche an.

Es kann mehrere Knoten in einem Cluster geben, die allerdings alle eine eigene Verbindung zum Master benötigen. D.h. es können keine Knoten “hintereinander” verkettet werden. Der Master ist allen Knoten übergeordnet.

Auf einem physikalischen Rechner kann neben einem Master ebenfalls auch ein Knoten arbeiten. Eine typische Cluster-Konfiguration für mapWOC ist z.B.:

- Rechner 1: Master, Redirector + Knoten A
- Rechner 2: Knoten B
- Rechner 3: Knoten C
- ...

mapWOC installieren

2.1 Systemvoraussetzungen

mapWOC nutzt extensiv folgende Technologien:

- kvm
- python
- pylons

Weiterhin wird eine Vielzahl an anderen freien Applikationen und Bibliotheken verwendet.

Für die Inbetriebnahme von mapWOC ist **Debian Squeeze 64bit** notwendig.

Um KVM verwenden zu können, ist eine **Hardwareunterstützung** zur Virtualisierung erforderlich. Zum Überprüfen, ob dies erfüllt ist, schauen sie nach den cpu-flags *vmx* oder *svm* in der Datei `/proc/cpuinfo`.

mapWOC wurde auf folgenden Systemen enwickelt und getestet:

1. DELL-Server PowerEdge 2970, 4 x Dual-Core AMD Opteron(tm) Processor 2222 SE
2. HP Compaq dx2450 Microtower, AMD Athlon 64 X2 5400B Dual Core 2800MHz
3. Teo-X Pro, Intel(R) Celeron(R) E3300 2500MHZ
4. MacMini

2.2 Installation

Die folgende Installationsanleitung bezieht sich auf **Debian Squeeze 64bit**. Nur für dieses System existieren derzeit mapWOC-Installationspakete.

1. Ergänzen Sie `/etc/apt/sources.list` um folgende Zeile:

   ```
   deb http://apt.intevation.org/ squeeze mapwoc
   ```

   und optional (wenn Sie an den Quellpaketen interessiert sind):

   ```
   deb-src http://apt.intevation.org/ squeeze mapwoc
   ```

2. Zum Verifizieren der Installationspakete aus dem hinzugefügten apt-Repository benötigen Sie den Intevations 'File Distribution Key':

```
gpg --keyserver hkp://keys.gnupg.net --recv-keys EC70B1B8
gpg --export EC70B1B8 | apt-key add -
```

(Diese und eine andere Methode sind auch unter http://apt.intevation.org beschrieben.)

3. Paket-Listen aktualisieren:

```
apt-get update
```

4. mapWOC-Pakete installieren:

```
apt-get install mapwoc-quick
```

Wichtig: Bei der Installation des mapwoc-quick Paketes werden Veränderungen am System vorgenommen, die nicht durch eine Deinstallation rückwirkend gemacht werden:

(a) Der Nutzer und Gruppe mapwoc werden angelegt und Daten in `/home/mapwoc/` hinterlegt.

(b) Eine Port-Weiterleitung in `/etc/rinet.conf` (80 -> 8123) wird eingerichtet.

(c) Registrierte HoneyClients werden nicht entfernt.

Die Installation dieses Paketes empfiehlt sich nur auf frischen Systemen, deren ausschließliche Nutzung ein Betrieb der mapwoc Software sein soll. Eine weniger intrusive Methode zur Installation eines mapwoc-Systems ist unter *"Ausführliche Konfiguration eines Einzelsystems"* zu finden. Es ist sicher zu stellen, dass der Nutzer mapwoc nach /tmp schreiben kann, gegebenenfalls sind Gruppen und Schreibeerlaubnisse anzupassen oder mount-Optionen zu ändern.

5. Reboot der Maschine durchführen:

```
reboot
```

Bei der Installation von mapwoc werden u.U. neue Kernel-Module installiert. Um sicherzugehen, dass diese auch geladen werden empfiehlt sich ein Neustart des Systems.

6. Sicherstellen, dass der sshd Schlüssel-basierte Authentifizierung erlaubt:

Dazu muss die Datei `/etc/ssh/sshd_config` die Zeile "PubkeyAuthentication yes" enthalten (Standard-Einstellung).

2.3 Demo-HoneyClient installieren

Zur Inbetriebnahme eines mapWOC-Systems werden zusätzlich KVM Gast-Images (*"mapWOC-Honey-Clients"*) benötigt.

Nutzen Sie für die Inbetriebnahme von mapWOC zunächst das frei verfügbare mapWOC-Demo-Image. Dabei handelt es sich um ein vorbereitetes, kleines Linux-KVM-Image auf Basis von SliTaz mit Firefox als Webbrowser.

Zur Installation des Demo-Images gehen Sie wie folgt vor:

1. Demo-Image von der mapWOC-Entwicklungsplattform herunterladen:

```
wget http://wald.intevation.org/frs/download.php/1135/slitaz_mapwoc-example-image.tar.gz
```

2. Archiv entpacken:

```
tar -xzf slitaz_mapwoc-example-image.tar.gz
```

3. Die Verzeichnisse *hcs* und *images* nach `/var/lib/mapwoc/node` kopieren:

```
cp -r hcs/ images/ /var/lib/mapwoc/node/
```

Um eigene KVM-Images anzulegen, lesen Sie später die Anleitung zur Erstellung eines virtuellen Honey-Clients im nächsten Kapitel.

2.4 mapWOC-Komponenten starten

Die vier mapWOC-Komponenten *mapwoc-redirector*, *mapwoc-master*, *mapwoc-node* und *mapwoc-gui* müssen nun nacheinander gestartet werden – möglichst jeweils auf einer separaten Konsole.

Wichtig: Jede Komponente muss als Nutzer *mapwoc* gestartet werden. Dieser Nutzer wurde vom mapWOC-Installationspaket passwortlos erstellt. Sie müssen also von einem höher-privilegiertem Nutzer (z.B. root) zu *mapwoc* werden.

Die Komponenten schreiben Log-Informationen auf die Standardausgabe stdout und in eine komponentenabhängige Logdatei (`mapwoc-redirector.log`, `mapwoc-node.log`, `mapwoc-master.log`).

1. Neue Konsole öffnen und den *mapwoc-redirector* starten:

   ```
   su - mapwoc
   mapwoc-redirector
   ```

2. Neue Konsole öffnen und *mapwoc-node* starten:

   ```
   su - mapwoc
   mapwoc-node
   ```

3. Neue Konsole öffnen und *mapwoc-master* starten:

   ```
   su - mapwoc
   mapwoc-master
   ```

4. Neue Konsole öffnen und die grafische Oberfläche starten:

   ```
   su - mapwoc
   paster serve mapwoc-gui/mapwoc-gui.ini
   ```

5. mapWOC-GUI im Browser öffnen:

   ```
   https://127.0.0.1:5000
   ```

   und mit dem initial eingerichteten Administrator-Nutzer anmelden:

   ```
   Nutzername: admin
   Passwort: siehe /home/mapwoc/mapwoc-gui/mapwocgui-admin-password
   ```

   Das Passwort sollte nach dem erstmaligen Einloggen geändert und die Passwort-Datei gelöscht werden.

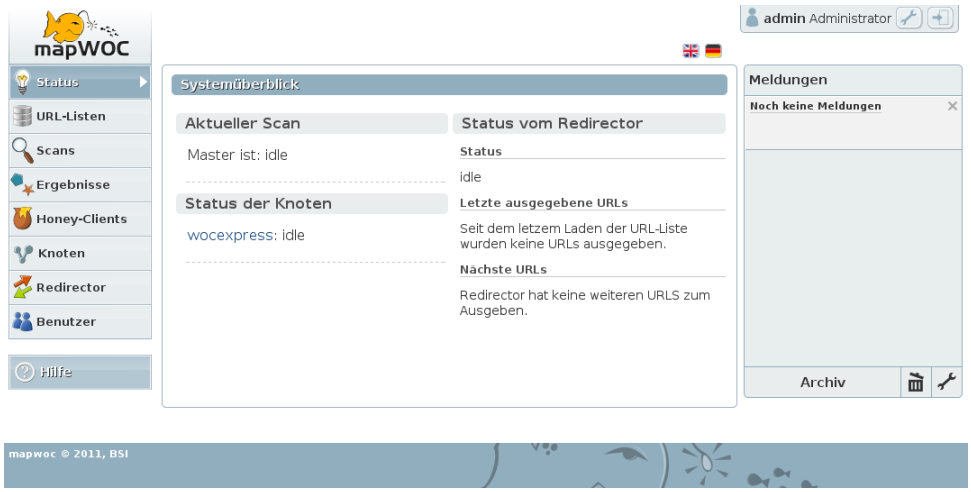
Bei Problemen wenden Sie sich bitte an die mapWOC-Entwicklermailingliste.

mapWOC benutzen

3.1 Die Benutzeroberfläche

mapWOC ist über eine Web-Oberfläche steuerbar, die sich in ein Drei-Spalten-Layout gliedert (siehe Screenshot):

1. Hauptmenü
2. Inhaltsbereich
3. Login- und Mitteilungsbereich



Aus Sicherheitsgründen verzichtet mapWOC vollständig auf JavaScript und andere aktive Inhalte.

In den nachfolgenden Abschnitten wird anhand der einzelnen Hauptmenüpunkte die Bedienung von mapWOC erläutert.

3.2 URL-Listen

Bei mapWOC konzentriert sich alles auf einzelne Webseiten (URLs). Ein Scan basiert auf einer vorher angelegten URL-Liste. URL-Listen bestehen aus beliebig vielen URLs.

Es gibt in mapWOC zwei Möglichkeiten URL-Listen zu **erstellen**:

1. **URLs manuell eingeben**

   Vergeben Sie einen frei wählbaren Namen für die Liste. Der optionale Kommentar hilft beim späteren Zuordnen der Liste.

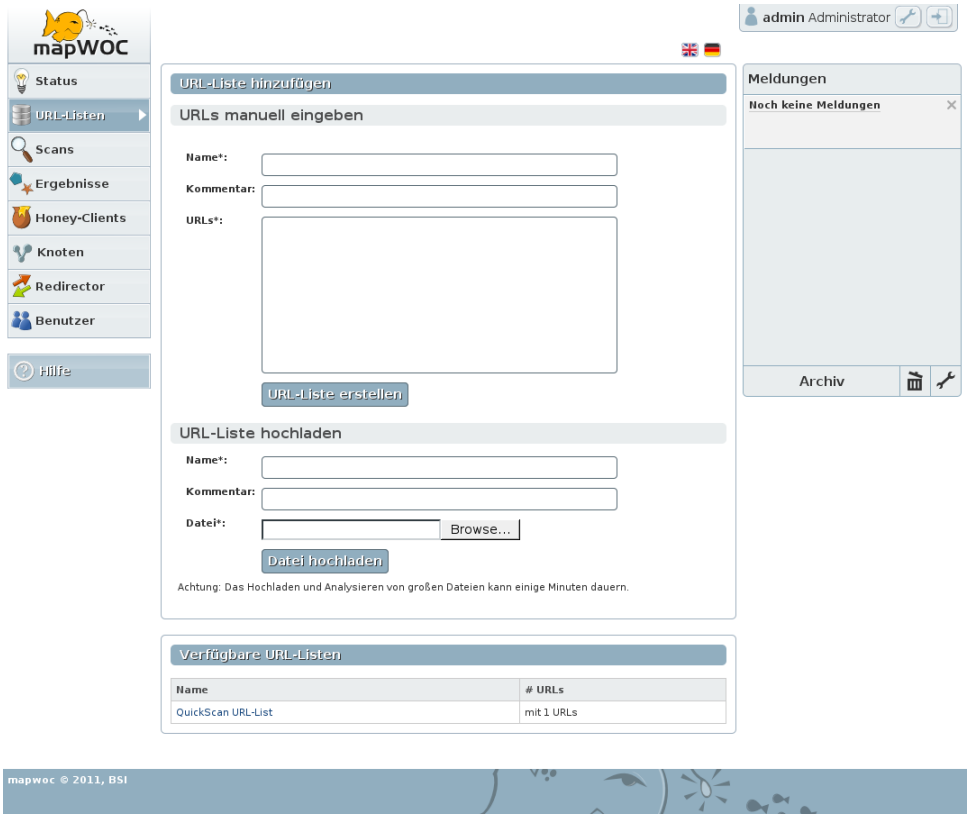
   Wichtig beim Eintragen der URLs ist die richtige Schreibweise, in der Form: `http://www.example.com`. mapWOC überprüft die Validität aller URLs beim Erstellen. Sollte bei einer URL z.B. ein `http://` fehlen, wird diese URL aus der Liste entfernt. Der Nutzer bekommt eine Fehlermeldung. Die Liste wird angelegt - jedoch ohne diese fehlerhafte Adresse.

2. **URL-Liste hochladen**

   URL-Listen lassen Sie auch aus vorliegenden Textdateien erstellen. Dazu die Datei auswählen und hochladen.

   > **Achtung:** Beachten Sie, dass beim Anlegen von sehr großen Listen das Hochladen, die Überprüfung und die Übertragung in die Datenbank einige Minuten dauern kann. Unterbrechen Sie diesen Prozess bitte nicht.

Im unteren Abschnitt der URL-Listen-Seite sind alle verfügbaren URL-Listen mit Anzahl der enthaltenen URLs aufgelistet. Ein Klick auf eine Liste zeigt weitere Details (Erstellungsdatum, URLs, Verknüpfung zu den Ergebnissen).



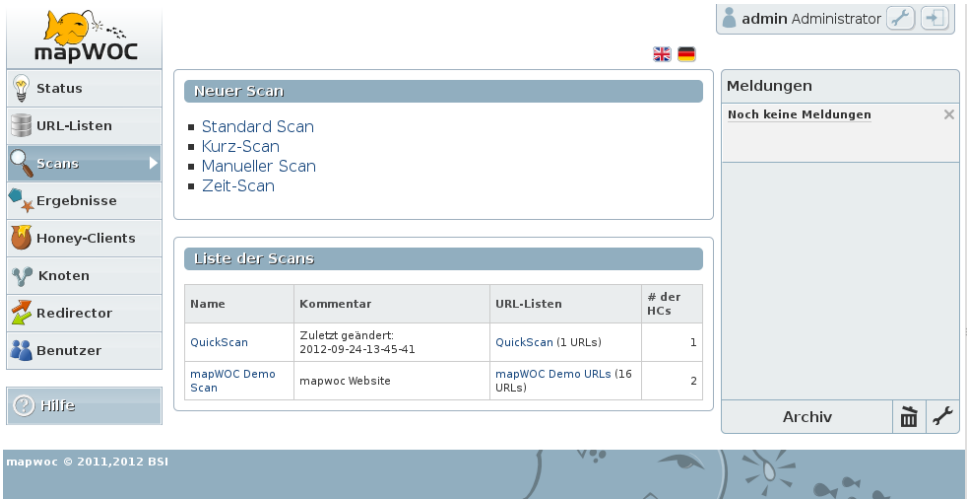
3.3 Scans

Ein Scan wird verwendet, um URLs in ausgewählten Browsern anzusurfen und den Netzwerkverkehr zu analysieren.

3.3.1 Neuen Scan anlegen

mapWOC bietet drei Arten von Scans:

1. **Standard-Scan**
2. **Kurz-Scan**
3. **Manueller Scan**
4. **Zeit-Scan**



zu 1. **Standard-Scan**

Ein Standard-Scan wird beendet, wenn die ausgewählten URL-Listen abgearbeitet sind.

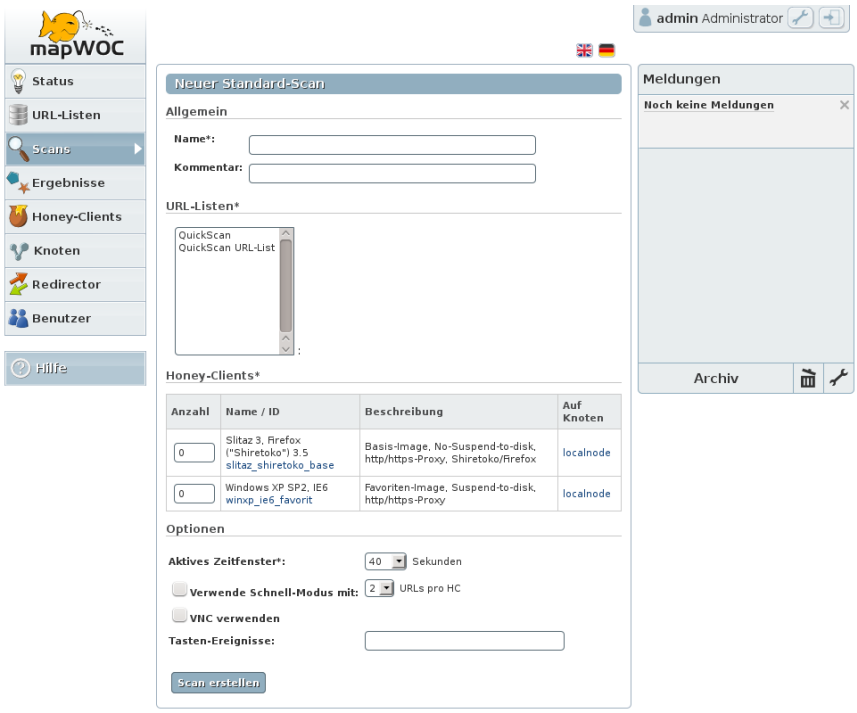
Zum Anlegen eines Standard-Scans sind folgende Angaben möglich (siehe Screenshot):

- Name vergeben
- Kommentar vergeben (optional)
- URL-Liste(n) auswählen
- Honey-Clients auswählen (Die Anzahl der parallel laufenden HCs ist hier erforderlich. Der Wert ist abhängig von den verfügbaren Hardwareressourcen und Softwarelizenzen.)
- Mehrere Optionen sind auswählbar:
- *Aktives Zeitfenster:* Das ist die Zeit vom Starten bis zum Abbruch des Honey-Clients. Voreingestellt: 40 Sekunden.
- *Schnell-Modus:* Dabei werden gleich mehrere URLs pro Honey-Client aufgerufen (ohne *Schnell-Modus* wird nur eine URL pro HC verwendet). Der Browser holt sich vom Redirector mehrere URLs und öffnet diese parallel in ebenso vielen Browserfenstern. Auswählbar sind 2 bis 5 URLs pro Honey-Client. Dieser Modus ist voreingestellt deaktiviert.
- *VNC:* Die laufende Scansitzung kann mit einem (externen) VNC-Viewer betrachtet werden. Der zugehörige VNC-Port wird für jeden HC auf der Statusseite während des Scans bereitgestellt. Die VNC-Hostadresse ist die Adresse des entsprechenden mapWOC-Knotens (Node), wo der HC läuft. Bei einem Nicht-Cluster-Betrieb ist diese identisch mit der Adresse von mapwoc-GUI. Dieser VNC-Modus ist voreingestellt deaktiviert.

- *Tasten-Ereignisse:* Nach einer bestimmten Zeit nach Start des virtuellen HCs kann das Ereignis *'Enter drücken'* (e) oder *'Gehe zur Startseite/Homepage'* (h) aufgerufen werden.

  Ein Beispiel: Die Eingabe von `10e15h` drückt automatisch nach 10 Sekunden Enter und nach weiteren 5 Sekunden (15 Sekunden nach Start) den Kurzbefehl für die Startseite (i.d.R. ALT+POS1).

Nach dem Anlegen eines Standard-Scans wird die Konfiguration auf einer Seite zusammengefasst. Zum Starten des Scans klicken Sie unter dem Abschnitt *Aktionen* auf *[Starten]*.



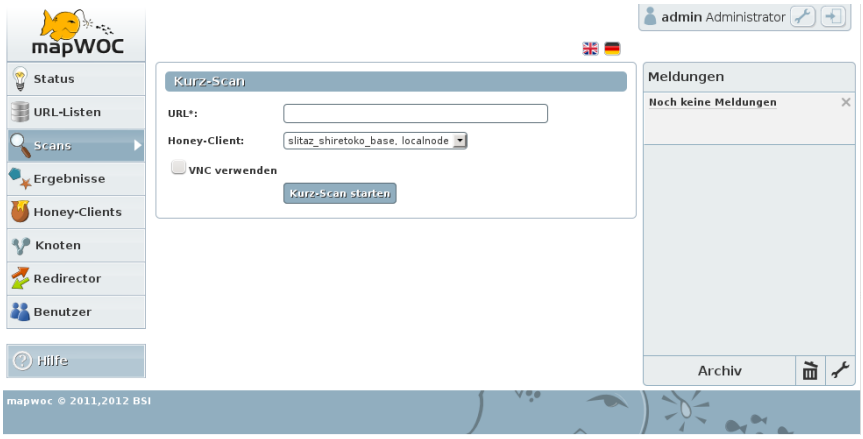
zu 2. **Kurz-Scan**

Ein Kurz-Scan wird beendet, nachdem die angegebene (Einzel-)URL angesurft wurde.

Zum Anlegen eines Kurz-Scans sind folgende Angaben erforderlich (siehe Screenshot):

- URL eingeben (in der Form `http://example.com`)
- Honey-Client auswählen
- Option *VNC*: Der Kurz-Scan kann mit einem (externen) VNC-Viewer betrachtet werden. Der zugehörige VNC-Port wird für den gewählten HC auf der Statusseite während des Scans bereitgestellt. Die VNC-Hostadresse ist die Adresse des entsprechenden mapWOC-Knotens (Node), wo der HC läuft. Bei einem Nicht-Cluster-Betrieb ist diese identisch mit der Adresse von mapwoc-GUI. Dieser VNC-Modus ist voreingesetllt deaktiviert.

Zum Starten des Scans klicken Sie auf *[Kurz-Scan starten]*.


mapWOC
admin Administrator
Status
URL-Listen
Scans
Ergebnisse
Honey-Clients
Knoten
Redirector
Benutzer
Hilfe
Kurz-Scan
URL*:
Honey-Client:
slitaz_shiretoko_base, localnode
VNC verwenden
Kurz-Scan starten
Meldungen
Noch keine Meldungen
Archiv
mapwoc © 2011,2012 BSI

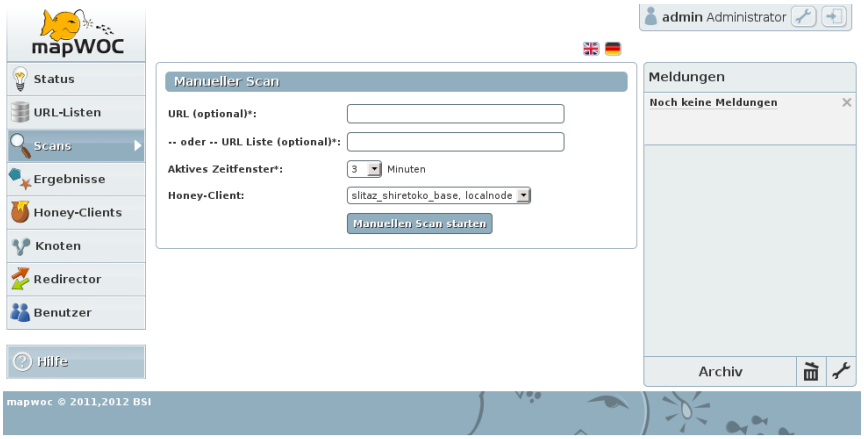
zu 3. **Manueller Scan**

Ein Manueller Scan wird beendet, wenn das ausgewählte Zeitfenster abgelaufen ist. Ein Manueller Scan dient nur zur manuellen Steuerung von **virtuellen** Honey-Clients. Native Honey-Clients können mit Manuellen Scans nicht betrieben werden; stattdessen können Zeit-Scan verwendet werden.

Zum Anlegen eines Manuellen Scans sind folgende Angaben erforderlich (siehe Screenshot):

- optional: URL eingeben *oder* URL-Listen auswählen
- Zeitfenster festlegen (voreingestellt: 3 Minuten)
- virtuellen Honey-Client auswählen

Zum Starten des Scans klicken Sie auf *[Manuellen Scan starten]*.



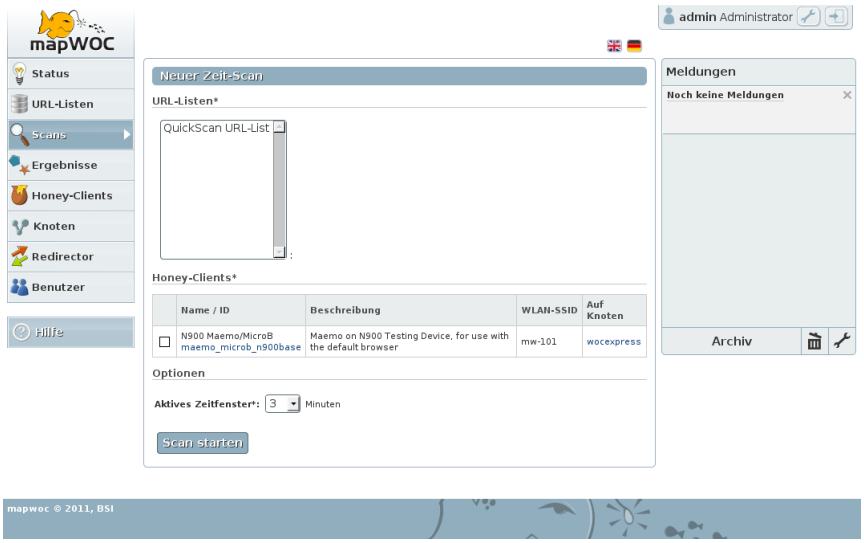
zu 4. **Zeit-Scan**

Ein Zeit-Scan wird beendet, wenn das ausgewählte Zeitfenster abgelaufen ist. Ein Zeit-Scan dient nur zur manuellen Steuerung von **nativen** Honey-Clients. Virtuelle Honey-Clients können mit Zeit-Scans nicht betrieben werden.

Zum Anlegen eines Zeit-Scans sind folgende Angaben erforderlich (siehe Screenshot):

- optional: URL-Listen auswählen
- nativen Honey-Client auswählen
- Zeitfenster festlegen (voreingestellt: 3 Minuten)

Zum Starten des Scans auf *[Scan starten]* klicken.



3.3.2 Scan-Liste

Im unteren Abschnitt der Scan-Übersichtsseite sind alle bisher angelegten Scans gelistet. Dabei werden zu jedem Scan die verwendeten URL-Listen (mit Gesamtsumme aller URLs) sowie die verwendeten Honey-Clients (mit Gesamtsumme aller HCs) angezeigt. Der Kurz-Scan ist voreingestellt und enthält immer genau eine URL.

Ein Klick auf einen Scan zeigt alle Konfigurationsdetails an. Hier kann ein Scan gestartet, gestoppt oder gelöscht werden.

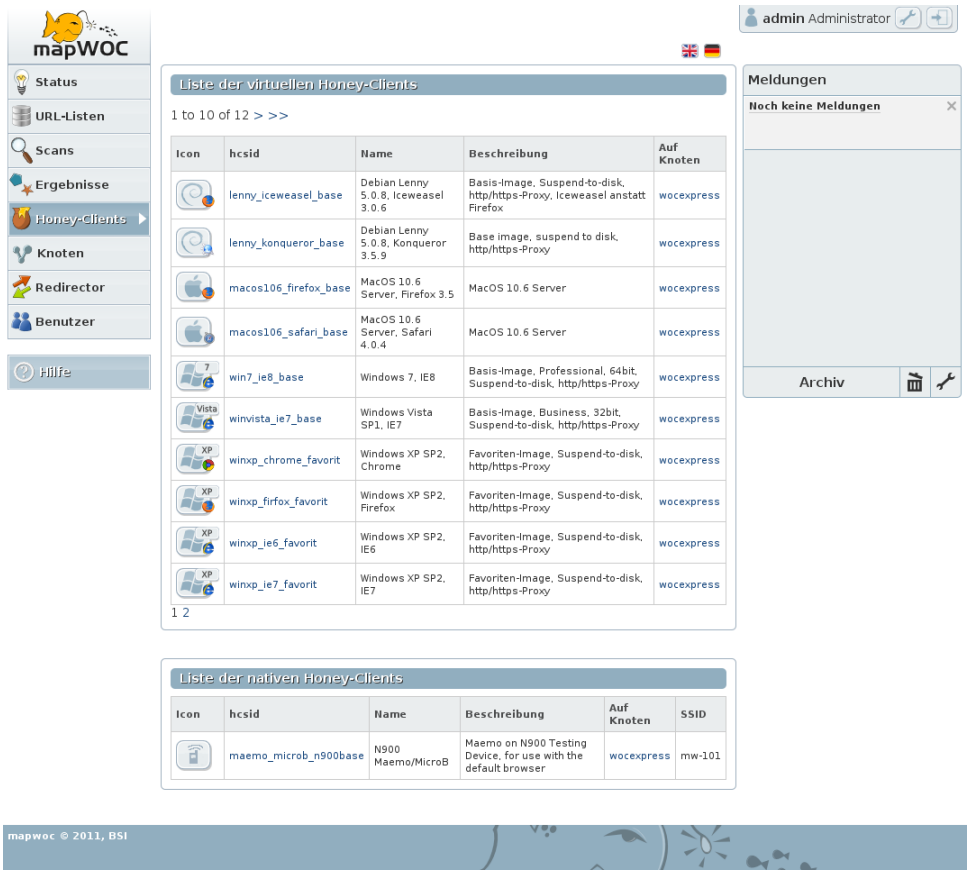
Wichtig: Das Bearbeiten eines Scans ist nicht möglich. Der Grund dafür ist, dass ein angelegter Scan mehrmals (hintereinander) durchgeführt werden kann. Um die Ergebnisse vergleichbar zu halten und eine eindeutige Referenzierung auf die Scankonfiguration zum Zeitpunkt des Scans zu gewährleisten, darf ein Scan nicht verändert werden. Möglich ist aber einen neuen Scan mit geänderter/ähnlicher Konfiguration anzulegen.

3.4 Honey-Clients

Ein Honey-Client (HC) ist ein Browsersystem, das zum Ansurfen von URLs in einem Scan genutzt wird.

Jedes beliebige Betriebssystem wird unterstützt, einzige Voraussetzung ist ein installierter Internetbrowser. mapWOC unterscheidet **virtuelle** und **native** Honey-Clients.

Der folgende Screenshot zeigt, wie so eine Liste mit Honey-Clients in mapWOC aussehen kann.



3.4.1 Virtuelle Honey-Clients

Virtuelle Honey-Clients sind Virtuelle Maschinen (VMn), die mit der Freien Virtualisierungslösung KVM erzeugt werden. Die Virtuellen Maschinen werden installiert, konfiguriert und mit einem gestarteten Browser abgespeichert (Snapshot) oder wahlweise in den Ruhezustand (suspend-to-disk) versetzt. Details zum Anlegen von KVM-Images für den Einsatz in mapWOC finden Sie im nächsten Abschnitt. Weiterführende Informationen zu QEMU/KVM bietet z.B. das deutschsprachige QEMU-Buch.

Der Snapshot/Ruhezustand eines virtuellen Honey-Clients wird von mapWOC geladen. Der Browser surft per Kurzbefehl die Redirector-Startseite an und holt sich eine URL ab. Nach Ablauf des aktiven Zeitfensters beendet mapWOC die VM wieder. Das KVM-Image ist anschließend wieder im Ausgangszustand (Snapshot/Ruhezustand). Änderungen werden nicht gespeichert.

Diese automatische Zurücksetzbarkeit ist der entscheidende Vorteil gegenüber nativen Systemen.

Honey-Clients sind über die Honey-Client-System-ID (hcsid) eindeutig bestimmt.

Virtuellen Honey-Client anlegen

Die folgende Anleitung (am Beispiel von Windows XP) hilft beim Anlegen eines KVM-Images, um es später in mapWOC als neuen virtuellen Honey-Client hinzuzufügen:

1. KVM-Master-Image erstellen (im qcow2-Format, mit einer 8 GB Festplatte):

```
kvm-img create -f qcow2 winxp-master.img 8G
```

2. Installation des Betriebssystems durchführen:

   ```
   kvm -cdrom windows-xp.iso -hda winxp-master.img -m 128M -vnc :1
   ```

   Dabei auf einer anderen Konsole einen VNC-Viewer starten, z.B.:

   ```
   vncviewer localhost:1
   ```

3. **Konfigurieren Sie das Netzwerk der VM wie folgt:**

   - IP: 10.0.0.4
   - Netmask: 255.255.255.0
   - Gateway: 10.0.0.1
   - DNS server: 10.0.0.3

   Starten Sie den Browser (hier IE6) und setzen Sie die Homepage auf folgende Adresse (muss mit der Angabe in der hc-Datei übereinstimmen; siehe Schirtt 8):

   ```
   http://10.0.0.1/winxp_ie6_demo
   ```

   Aktivieren Sie einen HTTP/HTTPS-Proxy für die Adresse:

   ```
   10.0.0.2:80
   ```

   Das vorbereitetes SSL-Zertifikat (mitmproxy) von mapWOC herunterladen und als vertrauenswürdige Stammzertifizierungsstelle im SSL-Zertifiaktsmanager importieren. (mapWOC integriert den mitmproxy als HTTP/HTTPS-Proxy.)

   Installieren Sie bei Bedarf weitere erforderliche Software.

   Wichtig: Deaktivieren Sie alle automatischen Updateprüfungen von Betriebssystem und installierter Software.

   Anschließend fahren Sie das Betriebssystem herunter.

4. KVM-Overlay-Image erstellen:

   ```
   kvm-img create -b winxp-master.img -f qcow2 winxp_ie6_demo.ovl
   ```

5. KVM-Overlay-Image starten:

   ```
   kvm -hda winxp_ie6_demo.ovl -m 128M -vnc :1
   ```

6. Browser starten (hier IE6).

7. Betriebssystem in den Ruhezustand versetzten.

   Bei Windows XP: *Start* > *Beenden* > *Herunterfahren* > *Ruhezustand* (Umschalttaste drücken)

   Das Betriebssystem speichert den Zustand in das Overlay-Image. Damit existiert nun ein Master- und ein Overlay-Image.

8. Damit mapWOC den neuen Honey-Client kennt, muss noch eine neue Konfigurationsdatei (hc-Datei) im `hcsdir`-Verzeichnis der Node angelegt werden.

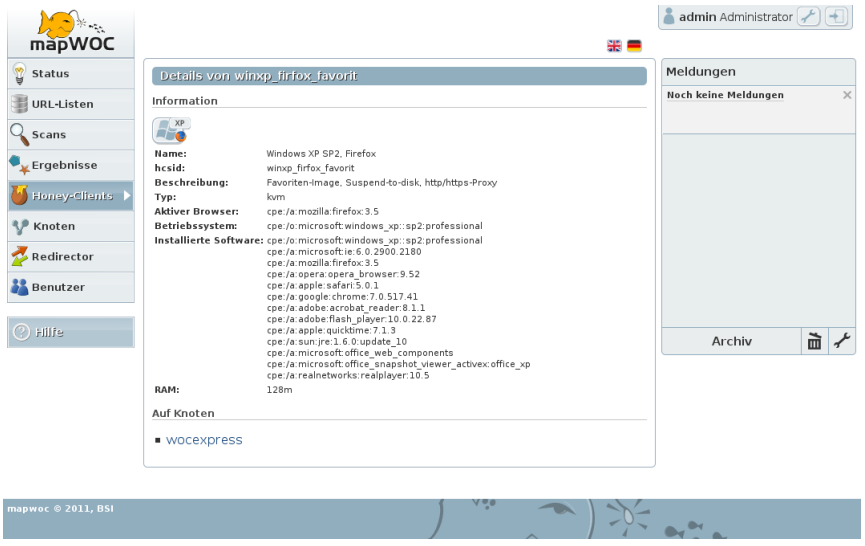
   Hier ein Beispiel für eine hc-Datei `winxp_ie6_demo.hc`:

```
[hc]
name=WindowsXP Demo Image IE6
description=Lots of other software on it
hcsid=winxp_ie6_demo
max_parallel=-1
controllable=true
active_browser=cpe://a:microsoft:ie6
installed_software=cpe:o:microsoft:etc

[control]
type=kvm
boots=true
image_master=/usr/share/mapwoc/images/winxp-master.img
image_file=/usr/share/mapwoc/images/winxp_ie6_demo.ovl
image_memory=128

[network]
own_ip=10.0.0.4
dns_ip=10.0.0.3
proxy_ip=10.0.0.2
gateway_ip=10.0.0.1
homepage=http://10.0.0.1/winxp_ie6_demo
```

9. Zuletzt muss nur noch die Node und der Master von mapWOC neu gestartet werden:

```
mapwoc-node
mapwoc-master
```

Der neue virtuelle Honey-Client *winxp_ie6_demo* erscheint nun in der Liste der Honey-Clients. Beim Klick auf diesen Eintrag lassen sich die Details anschauen (siehe Screenshot):



3.4.2 Native Honey-Clients

Native Honey-Clients sind Geräte, die nicht automatisiert von mapWOC gesteuert werden können (z.B. Mobilgeräte, Laptop). Automatisiert steuern kann mapWOC nur Virtuelle Maschinen (s.o.).

Native Honey-Clients bauen per WLAN eine Verbindung zu einem Access-Point auf, welcher am mapWOC-Server angeschlossen und eingerichtet ist.

Nativen Honey-Client hinzufügen

Um nun einen neuen nativen Honey-Client in mapWOC hinzuzufügen, befolgen Sie folgende Schritte:

1. MAC-Adresse des nativen Geräts über das Administrationsinterface des WLAN-Access-Points hinzufügen.
2. Eine neue hc-Konfigurationsdatei im `hcsdir`-Verzeichnis der Node anlegen.

   Hier ein Beispiel für ein Nokia N900 mit Maemo und MicroB-Browser in der hc-Datei `maemo_microb_n900.hc`:

   ```
   [hc]
   name=N900, MicroB
   description=Maemo
   hcsid=maemo_microb_n900
   max_parallel=1
   controllable=false
   active_browser=mapwoc-cpe:/a:nokia:microb
   os=mapwoc-cpe:/o:nokia:maemo
   installed_software=mapwoc-cpe:/o:nokia:maemo

   [control]
   type=native
   interface=vlan103
   image_memory=10

   [network]
   client_mac=c0:38:f9:ee:f0:ec
   wlan_ssid=mw-103
   proxy_ip=10.1.103.1:1066
   ```

3. Abschließend die Node und den Master von mapWOC neu starten:

   ```
   mapwoc-node
   mapwoc-master
   ```

Der neue native Honey-Client *maemo_microb_n900* erscheint danach in der Liste der Honey-Clients.

3.5 Ergebnisse

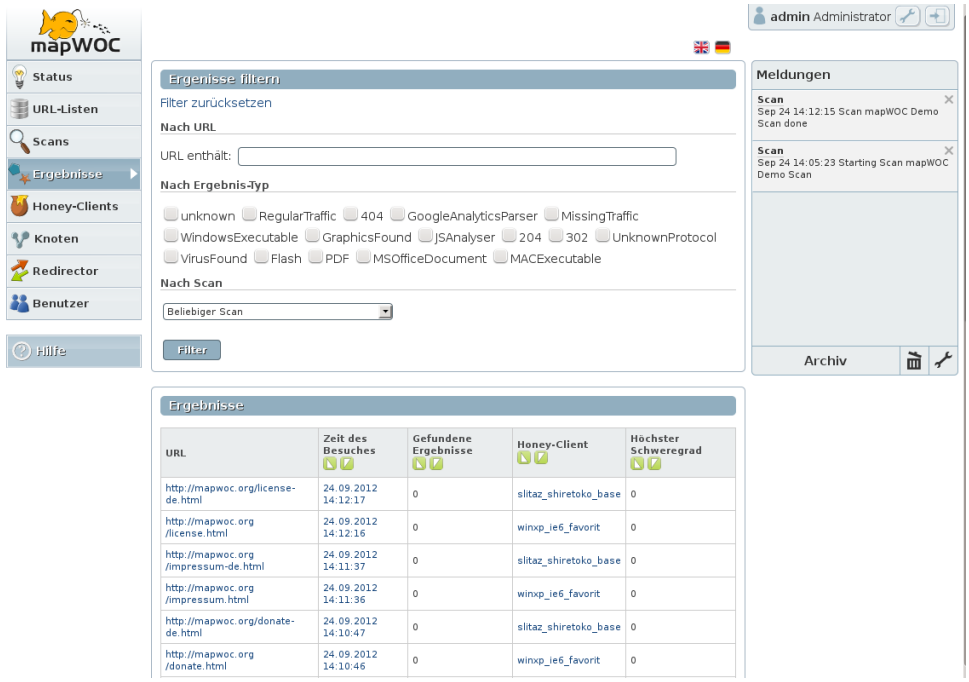
Die Ergebnisse aller Scans werden unter dem Menüpunkt *Ergebnisse* dargestellt.

Sie können diese Ergebnisse filtern:

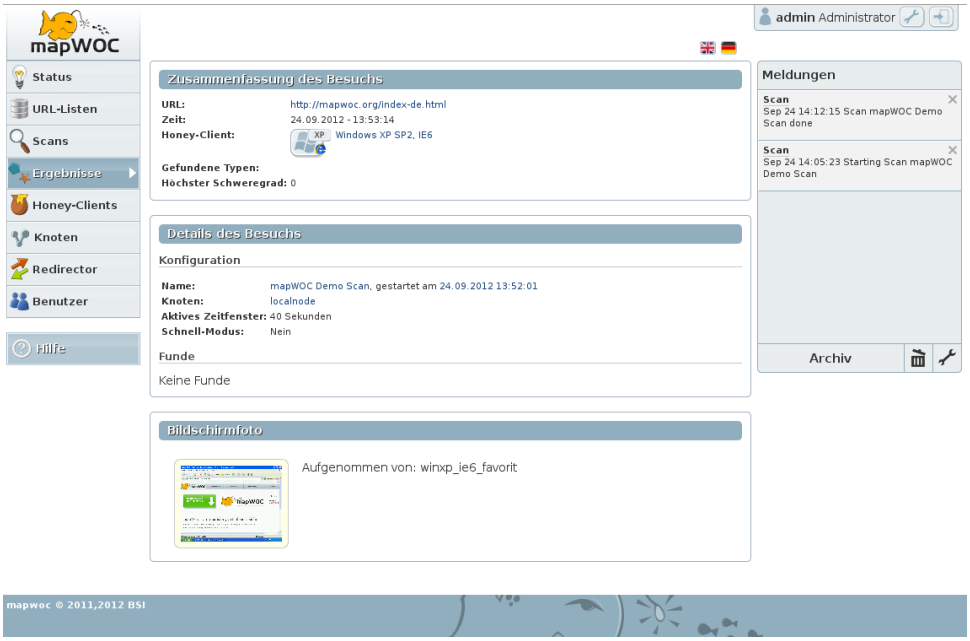
- nach einer Zeichenkette in einer URL (z.B. die Domain `example.com`),
- nach einem oder mehreren Ergebnis-Typen,
- nach einem vorhandenen Scan (aus den letzten 500 Scans).

Alle Filter können auch kombiniert werden.

Der folgende Screenshot zeigt eine Ergebnis-Übersichtsseite mit allen Filtermöglichkeiten:



Bei Auswahl eines “Besuchs” (d.h. ein Einzelergebnis, bei dem ein oder mehrere URLs analysiert worden sind) wird neben Scaninformationen des HC auch ein Bildschirmfoto der besuchten URL dargestellt. Anhand des Bildschirmfotos lässt sich so einfach erkennen ob oberflächlich an der Webseite etwas ungewöhnlich erscheint. Mögliche Funde auf der besuchten URL werden auf dieser Resultatseite in Tabellenform dargestellt – mit Angabe des gefundenen Typs und des jeweiligen Schweregrads. Der höchste Schweregrad dieser URL wird im oberen Bereich zusammengefasst.



3.5.1 Ergebnis-Typen

mapWOC analysiert den Netzwerkverkehr beim Besuch jeder URL. Dabei werden relevante, in einem Analyseskript festgelegte Dateien aus dem sogenannten *pcap*-Strom extrahiert und mit einem integrierten Virenscanner (voreingestellt ist *ClamAV*) untersucht. Jede extrahierte Datei wird als Fund bezeichnet. Jeder Fund wird bewertet, wie sicherheitskritisch diese Datei sein könnte, und in einen Schweregrad von 0 bis 999 eingeteilt: 0 ist ungefährlich, 999 bedeutet sehr kritisch.

Die nachfolgende Auflistung zeigt die verfügbaren Ergebnis-Typen mit ihrem jeweiligen Schweregrad - sortiert in absteigender Reihenfolge:

- unknown (999)
- VirusFound (900)
- MissingTraffic (600)
- WindowsExecutable (500)
- GoogleAnalyticsParser (500)
- MACExecutable (211)
- MSOfficeDocument (210)
- Flash (200)
- PDF (199)
- 404 (100)
- JSAnalyser (100)
- 204 (99)
- 302 (60)
- GraphicsFound (50)
- UnknownProtocol (001)
- RegularTraffic (000)

3.6 Redirector

Der Redirector ist zuständig für die Verteilung der URLs.

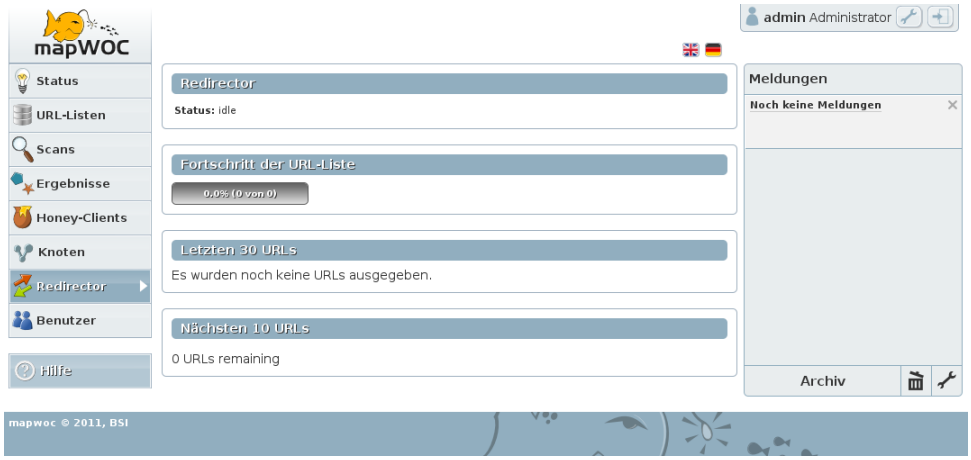
In jedem Browsersystem muss die URL des Redirectors als Startseite eingestellt werden:

```
http://10.0.0.1/<hcsid>
```

Also z.B.: *http://10.0.0.1/winxp_ie6_demo*

Durch Aufrufen dieser URL bekommt das Browsersystem vom Redirector die nächste URL aus der Liste geliefert. Wird der *Schnell-Modus* verwendet, werden vom Redirector gleich *n* URLs zurückgeliefert, die in *n* Browserfenstern geöffnet werden. (*n* ist die Anzahl der Seiten pro Honey-Client).

Die Redirector-Seite bietet während eines Scans eine Übersicht über den Fortschritt der verwendeten URL-Listen. Bei den letzten abgerufenen URLs wird zusätzlich der Honey-Client und der Zeitpunkt angegeben. Die nächsten URLs geben einen Ausblick, welche URLs als nächstes vom Redirector verteilt werden.


mapWOC
admin Administrator
Status
URL-Listen
Scans
Ergebnisse
Honey-Clients
Knoten
Redirector
Benutzer
Hilfe
Redirector
Status: idle
Fortschritt der URL-Liste
0.0% (0 von 0)
Letzten 30 URLs
Es wurden noch keine URLs ausgegeben.
Nächsten 10 URLs
0 URLs remaining
Meldungen
Noch keine Meldungen
Archiv
mapwoc © 2011, BSI

3.7 Benutzer

mapWOC unterteilt Benutzer entsprechend ihres Aufgabenprofils in drei Rollen:

1. **Gast**

   Ein Gast darf *keine* Änderungen vornehmen. Lediglich Informationen ansehen:

   - Systemstatus, Ergebnisse, URL-Listen, Scankonfigurationen, Honey-Clients, Knoten, Redirector und Meldungen ansehen
   - eigene Benutzereinstellungen ändern

2. **Normaler Nutzer**

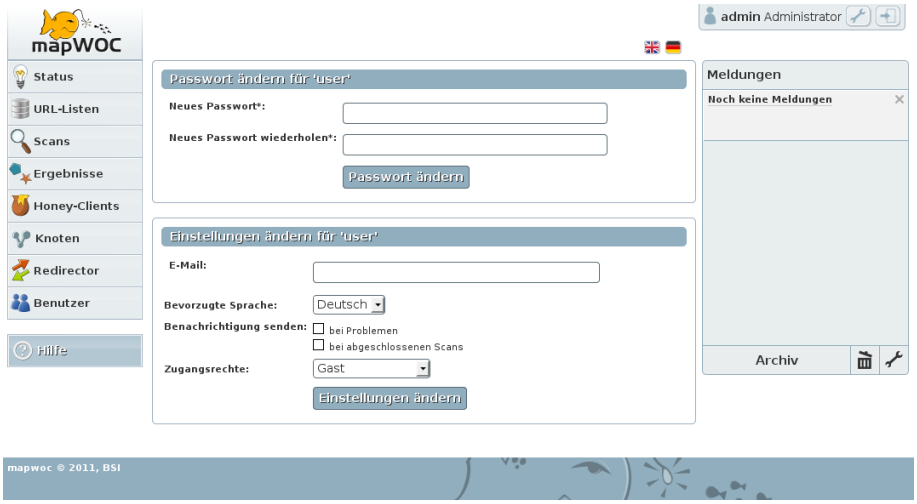
   Im Vergleich zum Gast darf der normale Nutzer zusätzlich:

   - URL-Listen verwalten
   - Scans anlegen/starten/stoppen

3. **Administrator**

   Im Vergleich zum normalen Nutzer darf der Administrator zusätzlich:

   - Benutzern verwalten
   - neue Knoten hinzufügen/entfernen
   - Systemeinstellungen ändern



mapWOC-Cluster aufbauen

Zum Aufbau eines mapWOC-Clusters sind (mindestens) zwei Rechner notwendig. Beide Maschinen werden aufgesetzt und das mapWOC-System sowie die Abbilder der VMn installiert. Die beiden Rechner werden an das Netzwerk angeschlossen und entsprechend konfiguriert.

Ein Rechner *R1* wird zum Master *M* ernannt. Zusätzlich wird auf *R1* ein Knoten *K1* eingerichtet. Master und Knoten laufen in der Regel unter zwei unterschiedlichen Benutzern auf *R1*.

Auf Rechner *R2* wird ein Knoten *K2* eingerichtet.

Für *M* wird ein SSH-Schlüsselpaar erzeugt und auf den beiden Knoten *K1* und *K2* bekannt gemacht (i.d.R. *~/.ssh/authorized_keys*). Ferner benötigt der Master zum autorisierten Zugriff auf die Knoten deren Fingerprints (*known_hosts*).

Über jeweils eine mapWOC-Konfigurationsdatei werden die beiden Knoten *K1* und *K2* auf dem Master registriert.

Nach dem Start aller mapWOC-Dienste sind *K1* und *K2* in der Benutzeroberfläche unter dem Menüpunkt *Knoten* sichtbar.

Um den Knoten *K2* aus dem Cluster herauszunehmen wird gegensätzlich gearbeitet: der SSH-Schlüssel des Masters *A* wird auf *K2* entfernt. Ferner muss auf dem Master *M* die Konfigurationsdatei *K2* entfernt werden. Nach Neustart der mapWOC-Dienste (master und node) besitzt *M* nur noch einen Knoten *K1*.

4.1 Konfiguration eines weiteren Knotens

Achtung: Die hier angegebene Konfiguration ist als Beispiel zu verstehen und sollte für den Produktivbetrieb sorgfältig mit vorhandener Infrastruktur und Sicherheitspolicy in Einklang gebracht werden.

1. Pakete installieren:

   ```
   apt-get install mapwoc
   ```

2. Nutzer einrichten:

   ```
   adduser --disable-password mapwoc
   usermod -a -G kvm mapwoc
   ```

   Bemerkung: Dieser Schritt entfällt, wenn das Paket mapwoc-quick installiert wurde.

   Wichtig: Folgende Schritte sind als Nutzer mapwoc auszuführen (sofern ausreichende Rechte vorhanden sind).

3. Sicherstellen, dass der sshd Schlüssel-basierte Authentifizierung erlaubt:

   Dazu muss die Datei `/etc/ssh/sshd_config` die Zeile "PubkeyAuthentication yes" enthalten (Standard-Einstellung).

4. Zugriff auf Resultate via öffentlichen Schlüssel (default: /var/lib/mapwoc/master/credentials/master_key.pub) vom master erlauben:

   ```
   mkdir ~/.ssh
   # master_key.pub vom Master auf Node kopieren.
   cat master_key.pub >> .ssh/authorized_keys
   rm master_key.pub
   ```

5. Dem Master den Node-host bekannt machen.

   Entweder /home/mapwoc/.ssh/known_hosts auf master-seite per Hand anpassen oder mit

   ```
   ssh -i /var/lib/mapwoc/master/credentials/master_key ip-of-node
   ```

   SSH diesen Eintrag schreiben lassen (die aufkommende Frage ist dann mit 'yes' zu beantworten).

6. Die Datei `/etc/mapwoc/mapwoc-node.config` anpassen, damit die Node auch nach außen lauscht:

   ctrl_interface=

   (Also das localhost nach dem "=" entfernen). Weitere Änderungen nur, wenn von den Default-Ports oder -Pfaden abgewichen werden soll.

7. HC-Dateien für die node hinterlegen:

   Eigene Clients hinzufügen oder *Demo-HoneyClient installieren*.

8. Auf der master-Seite ist eine node-configuration zu estellen:

   ```
   [remote-node]
   name=Remote Node
   description=This node runs on a different host than the master
   rsync_address=hostname
   ip=10.2.1.2
   port=18158
   ```

   (rsync_address und ip sind den Gegebenheiten anzupassen.)

1. Netzwerk

   Es ist sicherzustellen, dass der Redirector-Port (i.d.R. 80) auf den master weitergeleitet wird (z.B. mit ssh port-forwarding oder rinetd). Wenn das Paket mapwoc-quick installiert wurde, gibt es in `/etc/rinetd.conf` bereits einen Eintrag:

   ```
   localhost             80       localhost         8123
   ```

   Um die rinetd-Weiterleitung auf den master zu biegen, muss das zweite localhost (connectaddress) durch die IP des masters ersetzt werden. Da der redirector standardmäßig nur auf localhost lauscht, muss dies auf dem master geändert werden. Dort ist in der Datei `/etc/mapwoc/mapwoc-redirector.conf` der Wert von *redirector_interface*= entfernt werden (Zeichen hinter dem "=" entfernen). Beide Dienste (rinetd auf der node, mapwoc-redirector auf dem master) müssen neu gestartet werden, damit die Änderungen wirksam werden.

2. Die Node starten:

   ```
   mapwoc-node
   ```

   Bemerkung: Auf den zusätzlichen Knoten müssen weder mapwoc-redirector, mapwoc-master noch die GUI gestartet werden. Es muss allerdings sichergestellt sein, dass die HoneyClients den Redirector auf dem master erreichen.

3. Den Master neu starten:

```
# Auf dem master
mapwoc-master
```

mapWOC-Konfigurationsdateien

In diesem Kapitel werden die wichtigsten Konfigurationsdateien von mapWOC erläutert.

5.1 mapwoc-master

Der *mapwoc-master* steuert eine oder mehrere *mapwoc-nodes* und hört auf einer Adresse auf Befehle und Anfragen (z.B. von der mapwocgui).

/etc/mapwoc/mapwoc-master.config ist die Konfigurationsdatei für den *mapwoc-master* Befehl.

Hier ein Beispiel für so eine Konfigurationsdatei:

```
[config]
name=Master
address=localhost
node_config_dir=/var/lib/mapwoc/master/nodes/
credentials_path=/var/lib/mapwoc/master/key/master_key
ctrl_port=8558
work_dir=/var/lib/mapwoc/master/work/
db=/var/lib/mapwoc/db
redirector_port=8228
tolerance=101
#sslconfig=/var/lib/mapwoc/master/credentials

[notifications]
# Configuration of Mail notifications.

# Server used to send Mails with
smtp_server=mail.domain
smtp_port=25

# Mail address to send administrative notifications to
admin_mail=admin.mapwoc-demo@domain

# Address used by mapWOC as sender
sender_mail=MapWOC Demo <demo@mapWOC>
```

Alle Werte aus der Konfigurationsdatei lassen sich auch als Option über den Befehl *mapwoc-master* setzen. Hier ein Überblick aller verfügbarer **Optionen** (Auszug aus der Manpage von *mapwoc-master*, siehe auch `mapwoc-master --help`):

```
-v,--verbose
       Be verbose and chatty on stdout and logfile.

-q,--quiet
       Be quiet, only log warnings and errors.

--redirector_port
       Port to talk to when talking to redirector.

--ctrl_port
       Port to listen on for control commands.

--name Name of the master.

--address
       Interface to listen on.

--node_config_dir
       Directory in which to look for node config files.

--credentials_path
       Path to credentials to use when syncing results
       with nodes.

--work_dir
       Directory to place working data.

--db   Path to the database.

--tolerance
       Severity threshold (irrevertable).

--sslconfig
       Path to directory containing the {ca,cert,key}.pem files
       used for secured communication (talk plain text if not
       given).

--config
       Path to configuration file to read configuration from
       (or write to if called with --generate-config).

--generate-config
       Generate a configuration file.

-h,--help
       Show help and exit.

--version
       Show version and exit.
```

5.2 mapwoc-node

Die *mapwoc-node* hört auf Befehle des *mapwoc-masters*. Typischerweise resultieren diese Befehle im Starten und Stoppen von KVM-Prozessen und Analysieren des Netzwerkverkehrs von den Honey-Clients.

/etc/mapwoc/mapwoc-node.config ist die Konfigurationsdatei für den *mapwoc-node* Befehl.

Hier ein Beispiel für so eine Konfigurationsdatei:

```
[config]
ctrl_port=18158
ctrl_interface=localhost
hcsdir=/var/lib/mapwoc/node/hcs
result_dir=/var/lib/mapwoc/node/results
work_dir=/var/lib/mapwoc/node/work
num_analyser=4
max_gb_ram=4
analyser=/usr/bin/mapwocanalyse.pl
native_gate_script=/usr/lib/mapwoc/dumper_native_hc.sh
proxy_script=/usr/lib/mapwoc/mapwoc_redirect_script

#Port Range the node may use to start https proxies
proxy_ports=10000-11000

#sslconfig=/var/lib/mapwoc/node/credentials
```

Alle Werte aus der Konfigurationsdatei lassen sich auch als Option über den Befehl *mapwoc-node* setzen. Hier ein Überblick aller verfügbarer **Optionen** (Auszug aus der Manpage von *mapwoc-node*, siehe auch `mapwoc-node --help`):

```
-v,--verbose
       Be verbose and chatty on stdout and logfile.

-q,--quiet
       Be quiet, only log warnings and errors.

--ctrl_port
       Port to listen on for control commands.

--hcsdir
       Directory from which to read HC files.

--work_dir
       Directory in  which  to  work  and  store  temporary
       files.

--result_dir
       Directory in which to place results that are fetched
       from master.

--num_analyser
       Number of analyser processes to start in parallel.

--max_gb_ram
       Number of analyser processes to start in parallel.

--analyser
       The program to execute for analysis.

--analyser_conf
       Path to configuration file for analyser.
```

```
--native_gate_script
       The program to execute for control of native HCs.

--proxy_script
       Path to script for mitmproxy.

--proxy_ports
       The range of ports the node may use for ssl proxies

--sslconfig
       Path to credentials for secured communication.

--config
       Path to configuration file to read configuration from
       (or write to if called with --generate-config).

--generate-config
       Generate a configuration file.

-h,--help
       Show help and exit.

--version
       Show version and exit.
```

5.3 mapwoc-redirector

Der *mapwoc-redirector* startet zwei kleine Server. Auf dem einem Server werden HTTP-Anfragen mit HTTP-Redirects (Weiterleitungen) auf URLs einer Liste beantwortet. Der andere Server hört auf Kontrollbefehle, z.B. das Laden einer neuen URL-Liste.

/etc/mapwoc/mapwoc-redirector.config ist die Konfigurationsdatei für den *mapwoc-redirector* Befehl.

Hier ein Beispiel für so eine Konfigurationsdatei:

```
[config]
ctrl_interface=localhost
ctrl_port=8228
redirector_interface=localhost
redirector_port=8123
#sslconfig=/var/lib/mapwoc/redirector/credentials
```

Alle Werte aus der Konfigurationsdatei lassen sich auch als Option über den Befehl *mapwoc-redirector* setzen. Hier ein Überblick aller verfügbarer **Optionen** (Auszug aus der Manpage von *mapwoc-redirector*, siehe auch `mapwoc-redirector --help`):

```
-v,--verbose
       Be verbose and chatty on stdout and logfile.

-q,--quiet
       Be quiet, only log warnings and errors.

--urllist
       Path to file containing URLs.

-p,--redirector_port
       Port on which to listen for http requests

--redirector_interface
       Interface on which to listen for http requests.
```

```
--ctrl_interface
       Interface on which to listen for commands.

-c,--ctrl_port
       Port on which to listen for commands.

--sslconfig
       Path to credentials for secured communication.

--config
       Path to configuration file to read configuration from
       (or write to if called with --generate-config).

--generate-config
       Generate a configuration file.

-h,--help
       Show help and exit.

--version
       Show version and exit.
```

5.4 Honey-Clients

Honey-Client-Systeme (HCS) werden über hcs-Konfigurationsdateien der jeweiligen mapWOC-Node bekannt gemacht.

Das `hcsdir` – das Verzeichnis mit den Konfigurationsdateien – wird in der mapwoc-node.config festgelegt (Voreinstellung: `/var/lib/mapwoc/node/hcs`).

Hier ein Beispiel für eine hc-Datei `winxp_ie6_demo.hc`:

```
[hc]
name=WindowsXP Demo Image IE6
description=Lots of other software on it
hcsid=winxp_ie6_demo
max_parallel=-1
controllable=true
active_browser=cpe://a:microsoft:ie6
installed_software=cpe:o:microsoft:etc

[control]
type=kvm
boots=true
image_master=/usr/share/mapwoc/winxp-master.img
image_file=/usr/share/mapwoc/images/winxp_ie6_demo.ovl
image_memory=128

[network]
own_ip=10.0.0.4
dns_ip=10.0.0.3
proxy_ip=10.0.0.2
gateway_ip=10.0.0.1
homepage=http://10.0.0.1/winxp_ie6_demo
```

Hier ein Überblick der verfügbaren Parameter aus der Konfigurationsdatei eines Honey-Clients (HCs):

```
[hc]
name=
    Name des HCs.
```

```
description=
    Beschreibung des HCs.
hcsid=
    ID des HCs. Muss mit dem Dateinamen der HC-Konfigurationsdatei
    übereinstimmen.
max_parallel=
    Gibt an, wieviele Instanzen von diesem HC parallel gestartet
    werden können. Der Wert "-1" (= unendlich) wird für virtuelle
    HCs verwendet, "1" für native Geräte.
controllable=
    Gibt an, ob der HC von mapWOC steuerbar ist. Bei virtuellen
    HCs wird "true" verwendet, bei nativen HCs "false.
active_browser=
    Angabe des aktiven Browsers (in CPE-Form).
installed_software=
    Angabe der installierten Software (in CPE-Form). Mehrere
    Einträge durch ";" trennen.

[control]
type=
    Angabe des HC-Steuerungstyps. Mögliche Werte:
    - 'kvm' (KVM-Image)
    - 'kvm-mac' (KVM-Image mit MacOSX, erfordert spezielle
      KVM-Startparameter)
    - 'native' (nativer HC, wird manuell gesteuert)
boots=
    Diese Variable muss "true" sein, wenn sich das KVM-Image im
    Ruhezustand ("Supspend-to-disk") befindet. Andernfalls kann der
    'boots'-Parameter weggelassen werden.
image_master=
    Angabe des KVM-Master-Images (absoluter Pfad).
image_file=
    Angabe des KVM-Overlay-Images (absoluter Pfad).
image_snapshot=
    Angabe der KVM-Live-Migration-Snapshot-Datei (absoluter Pfad),
    sofern vorhanden.
image_memory=
    Angabe des durch den HC benötigten Arbeitsspeicher (in MB).
kvm_kernel=
    Angabe eines speziellen KVM-Kernel Parameters zum Starten.  Für
    KVM-MacOSX-Images wird z.B. benötigt:
    "kvm_kernel=/root/kvm-x/boot-osx-r327".  (Details siehe manpage
    von KVM, Option '-kernel').
interface=
    Bei Nutzung eines nativen HCs muss hier das VLAN-Interface
    angegeben werden, z.B.: 'vlan101'.

[network]
own_ip=
    IP-Adresse des HCs.
dns_ip=
    DNS-Adresse des HCs
proxy_ip=
    Proxy-Adresse des HCs
gateway_ip=
    Gateway-Adresse des HCs
homepage=
    Startseite des Browsers (Redirector mit hcsid), z.B.:
    'http://10.0.0.1/winxp_ie6_demo'
client_mac=
    MAC-Adresses des nativen HCs
wlan_ssid=
```

```
    SSID des zu verbindenen WLAN-Netzes für einen nativen HC.

[keyboard-shortcuts]
homepage=
    Optionaler Tastatur-Kurzbefehl, um die Browser-Startseite
    aufzurufen.  Voreingestellt ist bereits 'alt-home' (=Alt+Pos1).
    Anpassungen nur für bestimmte Browser/Betriebssysteme nötig.
    Z.B. für Safari unter MacOSX: '0xDB-shift-h' (=CMD+Shift+h)
```

5.5 Weitere relevante mapWOC-Dateien

Die nachfolgend genannten Pfade zu den Dateien sind Voreinstellungen von mapWOC. Änderungen der Pfade und Dateinamen sind über die zugehörigen Konfigurationsdateien von Master bzw. Node möglich. Die Angaben *[Master]* und *[Node]* sollen bei der Unterscheidung helfen, ob es sich um eine Datei des Masters oder der Node handelt.

- *[Master]:* Die **SQLite-Datenbank** liegt unter

  ```
  /var/lib/mapwoc/db
  ```

  Die Datenbank existiert nur auf dem Master.

- *[Node]:* Das **Arbeitsverzeichnis der Node** liegt unter:

  ```
  /var/lib/mapwoc/node/work/
  ```

  Hier werden die TCP-Dumps (im pcap-Format) von jedem KVM- und Proxy-Prozess für die Analyse abgelegt.

- *[Node]:* Das **Analysescript** liegt auf jeder Node unter:

  ```
  /usr/bin/mapwocanalyse.pl
  ```

  Dieses Skript analysiert Netzwerkverkehr aus den vorliegenden Dump-Dateien. Dabei werden gesuchte Dateien extrahiert (sofern vorhanden) und durch einen konfigurierten Virenscanner untersucht.

- *[Node]:* Die **Anti-Virenscanner-Konfiguration** werden in folgender Datei vorgenommen:

  ```
  /etc/mapwoc/antivir.cfg
  ```

- *[Node]:* **Analysierte Ergebnisse** eines jeden URL-Besuchs werden von der Node hier als tar.gz-Archiv abgelegt:

  ```
  /var/lib/mapwoc/node/results/
  ```

- *[Master]:* Der Master holt sich die bereitgestellten Ergebnisse von allen Nodes ab und legt sie alle gemeinsam in diesem **Master-Ergebnis-Verzeichnis** ab:

  ```
  /var/lib/mapwoc/master/
  ```

Ausführlichere Konfiguration eines Einzelsystems

Dieses Kapitel beschreibt die Konfiguration der automatischen Installation im Detail. Diese Konfiguration gilt auch für das Vorgehen bei älteren mapwoc-Installationen (bzw. bei manueller mapWOC-Installation ohne das Paket *mapwoc-quick*).

Für einen schnellen Einstieg, lesen Sie das Kapitel *"mapWOC installieren"*. Dort wird die automatische Installation (inkl. Standardkonfiguration) beschreiben.

Auf die Konfiguration eines gesicherten Verkehrs der Komponenten untereinender wird separat im Kapitel *"Setup CA"* eingegangen.

Beachten Sie, dass in diesem Handbuch eine Konfiguration zur Evaluation von mapWOC vorgestellt wird. Im Produktivbetrieb sollte mapWOC immer in Zusammenarbeit mit Sicherheitsbeauftragten und Systemadministratoren installiert werden.

Bemerkung: Die drei Komponenten *mapwoc-master*, *mapwoc-node* und *mapwoc-redirector* lassen sich über Konfigurationsdateien oder über die Befehle parametrisieren. Eine ausführliche englischsprachige Beschreibung lässt sich mit

```
mapwoc-node --help
```

oder

```
man mapwoc-node
man mapwoc-node.config
```

aufrufen.

Als Voreinstellung wird die Datei `/etc/mapwoc/mapwoc-node.config` geladen. Es kann jedoch auch eine andere Konfigurationsdatei geladen werden:

```
mapwoc-node --config /path/to/testing-node.config
```

Das hier beispielhaft an *mapwoc-node* gezeigte Verhalten gilt auch für *mapwoc-master* und *mapwoc-redirector* (vgl. auch Kapitel *"mapWOC-Konfigurationsdateien"*.).

Die drei Komponenten schreiben Log-Informationen auf die Standardausgabe stdout und in eine komponentenabhängige Logdatei (`mapwoc-node.log`, `mapwoc-master.log`, `mapwoc-redirector.log`).

6.1 Konfiguration im Detail

Eine Schritt-für-Schritt-Anleitung:

1. Neuen Nutzer anlegen:

```
adduser --disabled-password mapwoc
usermod -a -G kvm mapwoc
```

2. Umgebung anlegen:

```
mkdir -p /var/lib/mapwoc/master/credentials /var/lib/mapwoc/master/nodes/
mkdir -p /var/lib/mapwoc/master/work/
mkdir -p /var/lib/mapwoc/node/work /var/lib/mapwoc/node/results
chown -R mapwoc:mapwoc /var/lib/mapwoc
chmod 700 -R /var/lib/mapwoc/
```

   (Wenn die Standard-Einstellungen bezüglich der Datenspeicherung benutzt werden sollen.)

   Wichtig: Von nun an werden alle Aktionen als Nutzer mapwoc ausgeführt (*su - mapwoc*). Die Default-Konfigurationsdateien unter /etc/mapwoc müssen bei Bedarf für diesen Nutzer schreibbar gemacht werden.

3. Sich bekannt machen

   Der *mapwoc-master* wird in regelmäßigen Abständen mit dem rsync-Programm nach neuen Ergebnissen auf den Nodes sehen. Diese Art der Kommunikation ist mit ssh-Schlüsseln abgesichert. Ein solches Schlüsselpaar muss einmalig für den Master erstellt werden und der öffentliche Teil auf jeder Node als vertrauenswürdig hinterlegt werden. Weiterhin müssen dem master die hosts auf denen die Nodes laufen bekannt gemacht werden.:

```
mkdir /home/mapwoc/.ssh
chmod 700 /home/mapwoc/.ssh
```

   Das Schlüsselpaar wird (passwortlos) mit:

```
ssh-keygen -f master_key
```

   erzeugt und (entsprechend der Konfigurationsdatei) in `/var/lib/mapwoc/master/key/` hinterlegt (*Achtung:* restriktive 0700-Rechte für diesen Ordner mit *chmod 700 <dir>* setzen, ansonsten wird das Schlüsselpaar nicht benutzt!):

```
mv /home/mapwoc/master_key* /var/lib/mapwoc/master/credentials/
chmod 700 /var/lib/mapwoc/master/credentials/
```

   Nun wird ein Remote-Login mit diesem Schlüssel erlaubt:

```
cat /var/lib/mapwoc/master/credentials/master_key.pub >> ~/.ssh/authorized_keys
```

   Und der Host wird bekanntgemacht:

```
touch ~/.ssh/known_hosts
echo -n "localhost ssh-rsa " >> ~/.ssh/known_hosts
ssh-keygen -e  -f /etc/ssh/ssh_host_rsa_key | tail -n+3 | head -n-1 |
    awk '{printf "%s", $0}' >> ~/.ssh/known_hosts
```

   Alternativ zu letztem Schritt kann:

```
ssh localhost
```

   aufgerufen und die dann aufkommende Frage mit 'yes' beantwortet werden (danach mit Ctrl-C abbrechen).

4. Honey-Client-Dateien für die Node hinterlegen.

   Dazu müssen Sie vorher Honey-Clients angelegt haben. Informationen dazu finden Sie im nächsten Kapitel mapWOC benutzen.

5. Datei-Rechte oder Pfade in der Datei `/etc/mapwoc/mapwoc-node.config` anpassen.
6. Datei-Rechte oder Pfade in der Datei `/etc/mapwoc/mapwoc-master.config` anpassen.
7. Datei-Rechte oder Pfade in der Datei `/etc/mapwoc/mapwoc-redirector.config` anpassen.
8. Den *mapwoc-redirector* starten:

   ```
   mapwoc-redirector
   ```

9. Eine *mapwoc-node* starten:

   ```
   mapwoc-node
   ```

10. Dem *mapwoc-master* Zugang zu der *mapwoc-node* bekannt geben:

    Dazu im **node_config_dir** (über die Konfigurationsdatei `/etc/mapwoc/mapwoc-master.config` auf `/var/lib/mapwoc/master/nodes/` voreingestellt), eine Datei anlegen, z.B. `demo-node.config`:

    ```
    [demo-node]
    name=Demo Node
    description=This node runs on the same host as the master
    rsync_address=localhost
    ip=localhost
    port=18158
    ```

11. *mapwoc-master* starten:

    ```
    mapwoc-master
    ```

12. Die grafische Oberfläche aufsetzen und starten:

- Eine Ausgangs-Konfigurationsdatei erstellen.

  ```
  paster make-config "mapwocgui" mapwoc-gui.ini
  ```

- Die erstellte Datei `mapwoc-gui.ini` gegebenenfalls anpassen:

  Soll ein bereits existierendes Zertifikat genutzt werden, muss die Zeile

  ```
  ssl_pem =
  ```

  durch

  ```
  ssl_pem = /path/to/certificate
  ```

  ersetzt werden. Soll mit selbst-signierten Zertifikaten gearbeitet werden, wird

  ```
  ssl_pem = *
  ```

  eingetragen. Soll die ssl-Verschlüsselung ausgeschaltet werden, kann die Zeile entfernt oder durch ein "#" auskommentiert werden. Weiterhin kann der Port und das Interface konfiguriert werden. Mit

  ```
  host = 0.0.0.0
  port = 8080
  ```

wird die grafische Oberfläche über Port 8080 auf allen Netzwerkinterfaces zur Verfügung stehen. Außerdem sollten die Werte für **beaker.session.secret** und **authkit.cookie.secret** durch mehr oder weniger zufällige Werte ersetzt werden. Wurde mapwoc-master so konfiguriert, dass es auf einen anderen Port oder anderem Interface lauscht, ist der Wert **master.port** (**master.host**) anzupassen.

- Datenbank und initialen Nutzer einrichten:

```
paster setup-app mapwoc-gui.ini
```

- Die GUI starten:

```
paster serve mapwoc-gui.ini
```

CA für mapWOC aufsetzen

7.1 Hintergrund

Unkonfiguriert kommunizieren die mapWOC-Komponenten (z.B. *mapwoc-master* und *mapwoc-node*) **ohne Autentifizierung** und **unverschlüsselt**. Alle Komponenten unterstützen jedoch TLS-Authentifizierung und Verschlüsselung.

Für die Einrichtung einer CA wird in der Regel ein neues Paket benötigt, unter Debian Squeeze heißt dieses *gnutls-bin*.

Eine lesbare, generelle Einführung in das Thema asynchrone Verschlüsselung bietet z.B. das Gpg4win Kompendium.

7.2 Eine CA erstellen

Anstatt eine CA zu erstellen, kann natürlich auch eine bereits bestehende CA (oder Zertifikate) benutzt werden – siehe dazu weiter unten im Abschnitt "Einbindung der Zertifikate".

Nachfolgende Schritte sind in einem temporären und geschütztem Verzeichnis durchzuführen. Idealerweise findet dies nicht auf dem mapwoc-System sondern auf einem externen System statt.

Es können keine Passwörter verwendet werden.

1. Schlüsselpaar für die CA erzeugen:

   ```
   certtool --generate-privkey --outfile ca-key.pem
   ```

2. Zertifikat für die CA erzeugen:

   ```
   certtool --generate-self-signed --load-privkey ca-key.pem --outfile ca-cert.pem
   ```

   Bemerkung: Alle außer den folgenden Fragen können mit den Voreinstellungen beantwortet werden (Enter drücken):

   The certificate will expire in (days): 1000

   Does the certificate belong to an authority? (y/N): y

   Will the certificate be used to sign other certificates? (y/N): y

   Will the certificate be used for signing (required for TLS)? (y/N): y

   Is the above information ok? (Y/N): Y

3. Schlüsselpaar für den mapwoc-redirector erzeugen:

```
export CRDP=/var/lib/mapwoc/redirector/credentials/
certtool --generate-privkey --outfile $CRDP/key.pem
```

4. Zertifikat für den mapwoc-redirector erzeugen und signieren:

   ```
   certtool --generate-certificate --load-privkey $CRDP/key.pem \
            --outfile $CRDP/cert.pem --load-ca-certificate ca-cert.pem \
            --load-ca-privkey ca-key.pem
   ```

   Bemerkung: Alle außer den folgenden Fragen können mit den Voreinstellungen beantwortet werden (Enter drücken):

   Common name: mapwoc-redirector

   The certificate will expire in (days): 1000

   Is the above information ok? (Y/N): Y

5. Schlüsselpaar für den mapwoc-master erzeugen:

   ```
   export CRDP=/var/lib/mapwoc/master/credentials/
   certtool --generate-privkey --outfile $CRDP/key.pem
   ```

6. Zertifikat für den mapwoc-master erzeugen:

   ```
   certtool --generate-certificate --load-privkey $CRDP/key.pem \
            --outfile $CRDP/cert.pem --load-ca-certificate ca-cert.pem \
            --load-ca-privkey ca-key.pem
   ```

   Bemerkung: Alle außer den folgenden Fragen können mit den Voreinstellungen beantwortet werden (Enter drücken):

   Common name: mapwoc-master

   The certificate will expire in (days): 1000

   Is the above information ok? (Y/N): Y

7. Schlüsselpaar für die mapwoc-node erzeugen:

   ```
   export CRDP=/var/lib/mapwoc/node/credentials/
   certtool --generate-privkey --outfile $CRDP/key.pem
   ```

8. Zertifikat für den mapwoc-node erzeugen:

   ```
   certtool --generate-certificate --load-privkey $CRDP/key.pem \
            --outfile $CRDP/cert.pem --load-ca-certificate ca-cert.pem \
            --load-ca-privkey ca-key.pem
   ```

   Bemerkung: Alle außer den folgenden Fragen können mit den Voreinstellungen beantwortet werden (Enter drücken):

   Common name: mapwoc-node

   The certificate will expire in (days): 1000

   Is the above information ok? (Y/N): Y

9. Schlüsselpaar für die gui erzeugen:

   ```
   export CRDP=/var/lib/mapwoc/gui/credentials/
   certtool --generate-privkey --outfile $CRDP/key.pem
   ```

10. Zertifikat für den mapwoc-node erzeugen:

```
certtool --generate-certificate --load-privkey $CRDP/key.pem \
         --outfile $CRDP/cert.pem --load-ca-certificate ca-cert.pem \
         --load-ca-privkey ca-key.pem
```

Bemerkung: Alle außer den folgenden Fragen können mit den Voreinstellungen beantwortet werden (Enter drücken):

Common name: mapwoc-gui

The certificate will expire in (days): 1000

Is the above information ok? (Y/N): Y

11. Sichern des CA-Schlüssels:

    Der CA-Schlüssel ca-key.pem wird nun nur noch benötigt, wenn dem System neue Komponenten (z.B. eine neue Node) hinzugefügt werden sollen. Er ist von dem ausführenden System zu entfernen und separat gesichert aufzubewahren.

7.3 Einbindung der Zertifikate

Wichtig: Versichern Sie sich erneut, dass restriktive Rechte (nur Nutzer-lesbar) auf den u.g. Verzeichnissen und Dateien gesetzt sind.

1. CA-Zertifikat bekannt machen:

   ```
   cp ca-cert.pem /var/lib/mapwoc/redirector/credentials/
   cp ca-cert.pem /var/lib/mapwoc/master/credentials/
   cp ca-cert.pem /var/lib/mapwoc/gui/credentials/
   cp ca-cert.pem /var/lib/mapwoc/node/credentials/
   ```

2. Die Konfigurationsdateien der Komponenten anpassen:

   In den Konfigurationsdateien ist folgende Zeile auszukommentieren:

   ```
   #sslconfig=/var/lib/mapwoc/redirector/credentials
   ```

   wird dann zu

   ```
   sslconfig=/var/lib/mapwoc/redirector/credentials
   ```

   Der Pfad (/var/lib/mapwoc/redirector/credentials) ist für die anderen Komponenten entsprechend anzupassen (für den *mapwoc-master* z.B.: /var/lib/mapwoc/master/credententials), bzw. in den Vorgaben bereits angepasst.

   Bemerkung: In den credentials-Verzeichnis müssen sich drei nicht-leere Dateien befinden:

   key.pem (Schlüssel der Komponente)

   cert.pem (Zertifikat der Komponente)

   ca-cert.pem (CA-Zertifikat)

   Bemerkung: In der .ini Datei der GUI heißt die auszukommentierende Option nicht *sslconfig* sondern *master.sslconfig*, also z.B.

   master.sslconfig = /var/lib/mapwoc/gui/credentials

KAPITEL 8

Impressum