

Ein raumorientierter Sperrmechanismus für Internetdienste in Schulen

Daniel Jonietz
Fackelstraße 29
67655 Kaiserslautern
daniel@jonietz.de

Sommer 2002

Seit dem Einzug des Internet in die Schulen hegen Lehrer einen Wunsch: Den Zugang zum Internet zeitweise wieder sperren zu können. Die Gründe dafür sind vielfältig und reichen von Kostenkontrolle bis hin in den pädagogischen oder persönlichen Bereich, sie sollen hier nicht weiter analysiert werden. Statt dessen soll eine Möglichkeit zur Realisierung eines raumorientierten Sperrmechanismus aufgezeigt werden.

Einfache Maßnahmen zur Sperrung des Internet-Zugangs werden seit langem praktiziert und sind meist physikalischer Natur: Trennen wichtiger Verbindungen (z.B. durch Ziehen des ISDN- oder DSL- oder Netz-Steckers) oder Ausschalten zentraler Geräte (z.B. des Routers oder Hubs). Die Nachteile des Verfahrens liegen jedoch auch auf der Hand: Der Verbindung wird auf diese Art stets für alle Dienste getrennt und zudem oft für alle Räume, je nach Struktur des Netzes wird durch das Trennen einer Netzleitung vielleicht nicht nur die Verbindung zum Internet gelöst sondern auch die Verbindung zum Netzdrucker.

Eleganter ist eine Lösung, die eine raumweise (idealerweise vielleicht sogar eine rechnerweise) Kontrolle der Verbindung zu Internetdiensten ermöglicht. Der Lehrer kann dann frei entscheiden, welche Dienste des Internets den Schülern in seinem Raum zurzeit zur Verfügung stehen sollen ohne irgendwelche Nebenwirkungen befürchten zu müssen. Er allein behält die Kontrolle über das was in seinem momentanen Unterrichtsraum geschieht und beeinflusst dadurch nicht die Funktionsweise außerhalb.

Am Beispiel einer Realisierung am Burggymnasium Kaiserslautern soll eine Möglichkeit aufgezeigt werden, die in der Basisversion eine solche raumweise Kontrolle von Internetdienste ermöglicht. Diese Realisierung beschränkt sich in der hier beschriebenen Fassung erstmalig auf das Sperren des WWW: Einerseits ist das WWW einer der interessantesten Dienste des Internet, andererseits ist er aus technisch-administrativer Sicht besonders spannend, da hier direkt Rückmeldung an den Benutzer gegeben werden kann — der Benutzer fordert eine Seite an und erhält kontrolliert Rückmeldung das diese Seite nicht geholt wird weil der Arbeitsplatz des Benutzers für den Zugriff auf das WWW nicht freigegeben ist. Andere Dienste wie POP oder SMTP lassen sich dann weitgehend analog — meist einfacher — kontrol-

lieren, wobei die Möglichkeit der Rückmeldung nicht immer ohne weiteres gegeben ist.

Die Lösung soll folgenden Anforderungen genügen:

- Unabhängigkeit vom Betriebssystem auf Client-Seite
- Einfachste Bedienung für autorisierte Benutzer nach Authentikation
- Keine Nebenwirkungen, d.h. wirklich nur Sperren der beabsichtigten Dienste
- Erweiterbarkeit auf andere Dienste, insbesondere POP und SMTP.

1 Ist-Zustand am Burggymnasium

Die Schnittstelle zwischen Schulnetz und Internet stellt ein Linux-Router her, der im wesentlichen zwei Aufgaben erledigt: Masquerading (also Umsetzung der intern verwendeten privaten IP-Adressen auf eine öffentliche, routfähige Adresse) und Paketfilterung. Darüberhinaus stellt er dem Schulnetz DNS-Dienste zur Verfügung, was aber hier nur insofern von Bedeutung ist, als das Arbeitsplatzrechner keine externen Nameserver befragen müssen, sondern den schuleigenen DNS-Server nutzen sollen.

Auf dem zentralen Server (SuSE Linux 7.2) läuft ein Proxy-Server mit Cache (Squid), der über eigene Kontrolllisten (ACLs) und Positiv- und Negativlisten bestimmte Webseiten direkt sperrt. Prinzipiell ließen sich diese ACLs bei Bedarf einer Sperrung modifizieren, hier soll aber eine andere Lösung zum Einsatz kommen. Die Browser der Arbeitsplatzrechner brauchen und sollen also nicht direkt die entsprechenden Server kontaktieren, sondern immer den Proxy-Server ansprechen, der die Seiten dann nach einer augenscheinlichen Kontrolle durch die ACLs stellvertretend einholt. Der Rechner, auf dem der Proxy-Server läuft, bedient auch das Intranet. Dafür ist der Webserver Apache zuständig, er verarbeitet neben CGI-Skripten auch PHP-Seiten.

Das Internet ist immer verfügbar, einzelne Räume können — bei gleichzeitigem Verlust weiterer Funktionalität (Druckdienste) — durch physikalisches Trennen der Netze vom Internet gelöst werden.

Auf den Arbeitsplatzrechnern kommen verschiedene Windows-Versionen zum Einsatz.

2 Lösungsskizze

Die Grundidee ist recht einfach: Statt die Leitung zwischen Browser und Internet physikalisch zu trennen, wird ein Paketfilter installiert, der ein softwaremäßiges „weiches“ Trennen der Verbindung erlaubt. Gesteuert wird der Paketfilter über ein Webinterface. Diese Vorgehensweise ist äußerst flexibel: Prinzipiell ließe sich für jeden einzelnen Rechner jeder einzelne Dienst individuell sperren oder freigeben. Vorerst soll hier aber nur raumweise gesperrt werden.

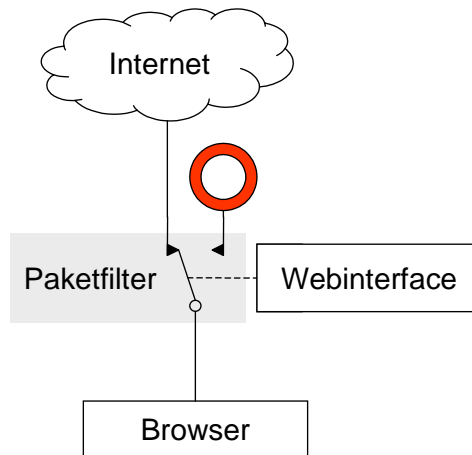


Abbildung 1: Grundidee der Internet-Sperre

2.1 Sperrmechanismus

Da alle Clients immer über den Proxy-Server auf das WWW zugreifen, kann dort über eine Filterung der eingehenden Seiten-Anforderungs-Pakete ein Sperrmechanismus greifen: Alle auf dem Server eingehenden Pakete werden kontrolliert und daraufhin untersucht, von welchem Rechner sie stammen. Handelt es sich um Pakete von einem Rechner, für den der Zugriff auf das WWW zurzeit gesperrt sein sollte, werden sie — nun ja, am einfachsten verworfen. Allerdings merkt der Benutzer dann erst nach einer gewissen Zeit (nach Ablauf eines Timeouts) das irgendwas nicht stimmt, die Fehlermeldung ist unspezifisch. Besser ist es, wenn der Benutzer gleich darüber informiert wird, das der Zugriff für ihn gesperrt ist. Daher werden die eingehenden Pakete nicht verworfen (DENY) oder zurückgewiesen (REJECT), sondern umadressiert (REDIRECT): Ein eigener Dienst wird als Antwort auf die Anfrage eine entsprechende Meldung zurückgeben. Die eigentlich an den Proxy-Server gerichtete Anfrage wird somit vom neuen Dienst bearbeitet.

2.2 Schaltmechanismus

Nun wird noch eine einfach zu bedienende Möglichkeit gebraucht den Sperrmechanismus in Gang zu setzen und zurückzusetzen. Im Sinne einer Öffnung des Systems und einer möglichst weitgehenden Befreiung kommt dazu ein Webinterface zum Einsatz, es benötigt auf Clientseite lediglich einen Browser und ist unabhängig vom auf den Arbeitsplätzen eingesetzten Betriebssystem.

Über dieses Webinterface kann ein autorisierter Benutzer den aktuellen Status des Raumes, in dem er sich gerade befindet, anzeigen lassen und ändern. Die dynamisch erzeugte Webseite zeigt genau den Raumnamen, den aktuellen Status und einen Schalter zum Wechsel zum Gegenstatus. Weitere Möglichkeiten (andere Räume einsehen, ...) werden vorerst bewußt nicht angeboten.

Die über das Webinterface angegebenen Befehle werden dann vom System verarbeitet und bewirken die Umschaltung der Wirkungsweise der Paketfilterung.

3 Vorarbeiten am Router

Damit niemand am Proxy-Server „vorbeisurfen“ kann, wird dem Paketfilter des Routers eine neue Filterregel hinzugefügt. Sie soll bewirken, dass der Router nur noch Pakete vom Server annimmt und alle Pakete, die von irgendwelchen Arbeitsplätzen kommen, verworfen werden. Damit wird tatsächlich ein Proxy-Zwang realisiert:

- Initialisierung (-F) der Filterregeln für die **forward**-Kette:
`ipchains -F forward`
- Setzen der Standard-Policy (-P) auf DENY (also alle Pakete dieser Kette werden verworfen):
`ipchains -P forward DENY`
- Hinzufügen einer neuen Filterregel (-A) für die **forward**-Kette: Alle Pakete die vom Server kommen (-s 192.168.0.1), egal wo sie hin sollen (-d 0/0), egal welches Protokoll sie nutzen (-p ALL) werden auf die Kette verwiesen, die sich um Masquerading kümmert (-j MASQ):
`ipchains -A forward -s 192.168.0.1 -d 0/0 -p ALL -j MASQ`

Sollen für die Arbeitsplatzrechner irgendwelche anderen Dienste (z.B. ein externer Mail- oder DNS-Server) direkt erreichbar sein, so müssen diese ebenfalls an dieser Stelle freigegeben werden.

4 Technische Realisierung

4.1 Bereitstellung der Fehlerseiten

Um dem gesperrten Anwender bewußt zu machen warum er keine Webseiten erhält und ihn nicht ewig auf Timeouts warten zu lassen soll er explizit informiert werden: Für diesen Arbeitsplatz ist der Zugriff auf das WWW zurzeit gesperrt.

Dazu wird neben dem Webserver Apache, der auf dem Standard-Port 80 arbeitet, ein zweiter Webserver, tHTTPd, auf Port 8080 aufgesetzt. Das Intranet-Verzeichnis dieses Webserver ist aber leer, so dass er bei jedweder Anfrage die Fehlermeldung: „Seite nicht gefunden“ zurückliefern muss. Der Text dieser Fehlermeldung läßt sich nun bei tHTTPd leicht abändern, so dass in Zukunft für den eigentlich verursachten Fehler „Seite nicht gefunden“ die Meldung: „Von diesem Rechner aus ist der Zugriff auf das Internet zurzeit gesperrt“ erscheint.

4.2 Umlenken der Anfragen

Für jeden Rechner des zu sperrenden Raumes muss der Paketfilter des Servers, auf dem der Proxy-Server angesiedelt ist, eine neue Filterregel erhalten, die Regel wird wieder gelöscht, wenn der Raum freigegeben werden soll. Diese Regel soll bewirken, dass die Anfrage des Clients nicht an den Proxy-Server gelangt, sondern an den Fehlermeldungsserver tHTTPd geleitet wird. Im Detail wird das Paket nicht an den Port 3128 (auf dem der Proxy Squid lauscht) geleitet, sondern auf Port 8080 umgebogen, damit tHTTPd eine Fehlermeldung ausgibt.

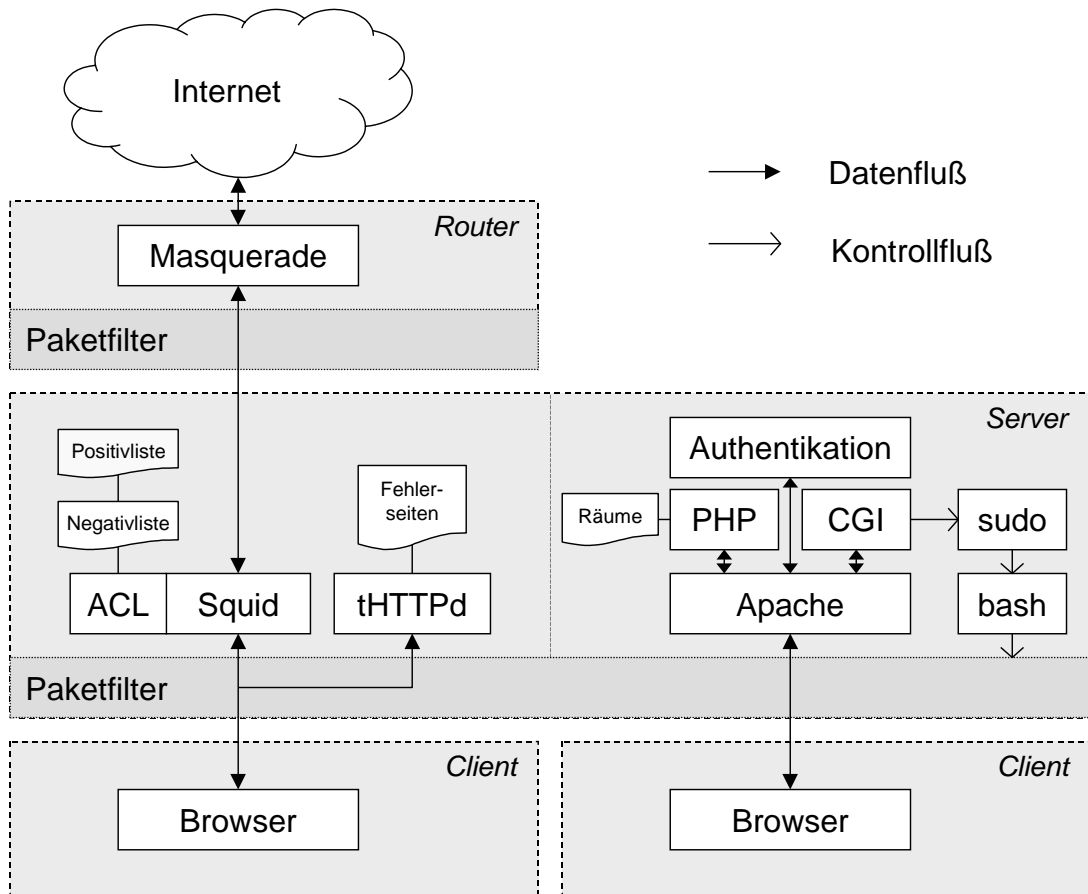


Abbildung 2: Schematischer Überblick: Der Block *Server* zeigt links Proxy-Server und Fehlermeldungssever, rechts die Module die das Verhalten des Paketfilters steuern und somit letztendlich den Weg festlegen, den die Pakete nehmen. Der linke Client zeigt einen Schüler-Arbeitsplatz, der rechte Client einen Nutzer des Webinterfases (Lehrer).

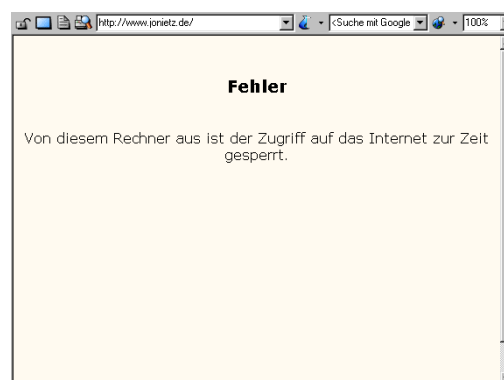


Abbildung 3: Versuch eines gesperrten Rechners auf `www.jonietz.de` zuzugreifen.

- Sperren (besser also: Umlenken)
`ipchains -A input -s 192.168.0.100 -d 192.168.0.1 3128 -p tcp ↔`

```
-j REDIRECT 8080
```

- Freigeben (die Umlenkung aufheben)

```
ipchains -D input -s 192.168.0.100 -d 192.168.0.1 3128 -p tcp ←  
-j REDIRECT 8080
```

Die Befehlszeilen bewirken das Anlegen bzw. Löschen der Filterregel, die besagt: Alle Pakete, die von dem Rechner mit der Adresse 192.168.0.100 kommen und an den Rechner 192.168.0.1 auf Port 3128 gerichtet sind und das tcp-Protokoll verwenden werden an Port 8080 umgeleitet. Die Umleitung geschieht transparent, der Anwender merkt nicht dass seine Anfrage nicht länger vom Proxy-Server Squid sondern vom Fehlermeldungs-Server tHTTPd bearbeitet wird.

Für andere Dienste wie SMTP oder POP ist keine Umleitung notwendig, es reicht dann ein REJECT oder DENY aus.

Ein Shell-Skript `icontrol` nimmt das Umschalten für alle Rechner des Raumes vor, der Raumname wird als Parameter übergeben. Für jeden Raum liegen in einem eigenen Verzeichnis Dateien die den Raumnamen als Dateinamen tragen und zeilenweise die IP-Adressen der Rechner in diesem Raum beinhalten.

Das Skript legt auch eine Statusdatei an, die den momentanen Status des Raumes festhält: Dazu wird dem Raumnamen als Trennzeichen ein Punkt und dann ein `closed` oder `opened` angehängt. Ist der Raum B04 gesperrt, so existiert eine Datei `B04.closed`. (Dies kann natürlich im Fehlerfall Probleme aufwerfen; Insbesondere müsste nach einem Neustart des Servers der Paketfilter mit dem Zustand auf Platte abgeglichen werden. Auch könnte es nicht schaden, wenn die Wirkung des Skriptes atomar wäre.)

Dieses Skript darf natürlich nur vom Superuser `root` aufgerufen werden, da es direkt die Paketfilterregeln manipuliert.

Damit ist die grundlegende Funktionalität gegeben: Das Skript wechselt bei jedem Aufruf den Status des Raumes: Von gesperrt zu freigegeben und umgekehrt. Auch beim Aufruf explizit mittels der ACLs gesperrter Seiten wird nur die Meldung, das der Zugang gesperrt sei, ausgegeben. Der Proxy selbst ist für gesperrte Rechner nicht erreichbar, wohl aber das Intranet.

4.3 Benutzerinteraktion: Webinterface

Das Webinterface besteht im Wesentlichen aus zwei dynamisch erzeugten Webseiten und einem CGI-Skript. Die Haupt-Webseite wird mittels PHP erzeugt: Der momentane Status wird über die vom Skript `icontrol` angelegten Statusdateien ermittelt und ausgegeben, entsprechend ein Schalter „freigeben“ bzw. „sperren“ erzeugt. Sobald ein Benutzer den Schalter betätigt wird das CGI-Skript `icontrol.cgi` aufgerufen, das eigentlich nichts anderes tut als die Parameter zu ermitteln, den eigentlichen Aufruf zum Statuswechsel startet und den Benutzer darüber informiert, dass die Dinge im Gange sind.

Da prinzipiell auch mehrere Personen in einem Raum das Skript zeitgleich nutzen können kann der angezeigte Status schnell veraltet sein, daher aktualisiert sich die Webseite periodisch selbstständig.

Bleibt nur noch ein Problem: Der Webserver läuft unter der unprivilegierten Kennung `wwwrun` und darf daher das Skript `icontrol` nicht aufrufen. Der Befehl

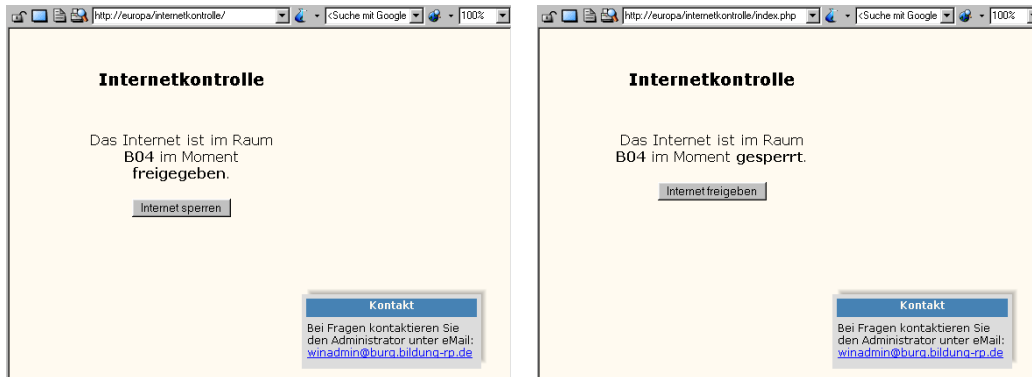


Abbildung 4: Das Webinterface, aufgerufen in einem gesperrten Raum (links) und einem freigegebenen Raum (rechts).

`sudo` kann hier weiterhelfen: Wir erlauben `wwwrun` per `sudo` das Skript `icontrol` ohne Passwortabfrage mit Superuser-Rechten laufen zu lassen. Dazu ist der Eintrag:

```
wwwrun ALL=NOPASSWD: /root/icontrol
```

in der Datei `/etc/sudoers` notwendig, er kann mittels `visudo` eingegeben werden.

Der Kern von `icontrol.cgi` ist dann die Zeile:

```
sudo /root/icontrol ${QUERY_STRING:5:8}
```

die über `sudo` das Skript `icontrol` aufruft und ihm die aufbereiteten Parameter übergibt. Dazu extrahiert es aus der Variable `QUERY_STRING` die Zeichen nach dem fünften bis einschließlich des achten.

4.4 Benutzerautorisierung

Die Webseite als Kontrollzentrum der WWW-Verbindung soll nun sicherlich nur einem bestimmten Personenkreis zugänglich sein, der Aufruf an die Kenntnis eines Passwortes gebunden sein. Da die zentrale Benutzerdatenbank auch Benutzer aufführt, denen der Zugang zum Webinterface nicht gestattet sein soll, kann nicht einfach auf die `/etc/shadow`-Datenbank zurückgegriffen werden. Auch die Idee eines einzelnen Zugangs mit nur einem Passwort, das dann an alle Berechtigten verteilt wird, kann getrost verworfen werden.

Statt dessen wird eine eigene Datenbank eingerichtet, die die autorisierten Benutzer enthält. Damit keine manuelle Verwaltung selbiger notwendig ist, wird sie automatisiert und an die Benutzerdatenbank gekoppelt: Ein Skript ermittelt alle Lehrer und Administratoren und kombiniert ihre Benutzernamen mit den entsprechenden Passwörtern aus dem Shadow-System. Dieses Skript wird per `crontab` regelmäßig aufgerufen und gleicht die Benutzerdatenbank mit der Datenbank die zur Nutzung des Webinterfaces berechtigt ab. Diese Lösung hat den weiteren Vorteil, dass Nutzer sich nur eine Kombination von Benutzername und Passwort merken müssen: Das Passwort zur Anmeldung am System erlaubt auch die Nutzung besonderer Dienste.

Die eigentliche Authentikation geschieht dann über den Apache-Standardmechanismus und die `.htaccess`-Datei.

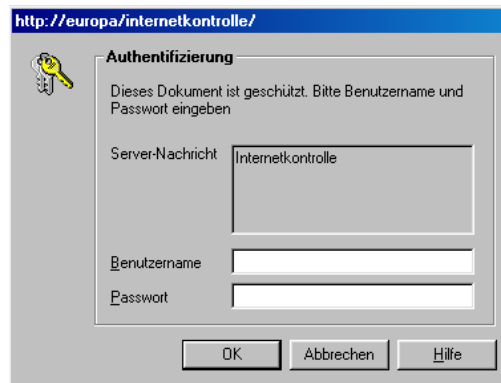


Abbildung 5: Aufforderung zur Authentikation vor der Nutzung des Webinterfaces.

4.5 Einbindung in das Gesamtsystem

Um Lehrern einen einfachen Zugriff auf den Verbindungsstatus ihres Raumes zu liefern wird auf dem Desktop für alle Lehrer eine Verknüpfung auf die entsprechende Webseite gelegt. Mit einem Doppelklick ist damit die Internetkontrolle möglich.

5 Ausblick

Die beschriebene erste Realisierung hat noch Haken und Ecken, die sich aber allmählich lösen lassen. Verfeinerungen sind viele denkbar, angefangen von einer Übersicht über die Räume und ihren Verbindungs-Status für besonders autorisierte Nutzer oder eine detailliertere Sperrung anderer Dienste. Obiger Text beschreibt nur die Sperrung des Standard-WWWs, Dienste wie POP oder SMTP bleiben offen. Zu Ergänzen wäre das Skript `icontrol` und ggf. die Webseite. Die Status-Dateien könnten genaue Informationen aufnehmen, welche Dienste jetzt gesperrt oder freigegeben sind. Im Moment arbeiten sie lediglich über ihr Vorhandensein.

Auf der anderen Seite zeigt dieser Ansatz ein hohes Maß an Flexibilität und einfacher Handhabung. Weitere Räume (oder auch einzelne Rechner) müssen dem System nur mitgeteilt werden. Der Lehrer muss nicht am System angemeldet sein, um das Webinterface zu nutzen, dies ist auch mit einem unprivilegierten Zugang möglich. Erst für den Aufruf des Webinterfaces ist ein autorisierter Zugang erforderlich.

Materialien

Die vollständigen Dateien stehen unter www.jonietz.de/schule/ zum Download bereit.