

BuzzerStar PROXY V1.0. –Anforderungsprofil

2013-10-31 / v1.2 / Enger

Allgemein:

- Handy app soll xml config standardwerte mitbringen
- Proxy hat aktuelle xml config liste vorliegen
- Proxy überträgt xml config liste immer mit ans handy (oder 1 mal am tag beim ersten app start)
- Priorität:
 1. Proxy XML Config ,die mitgesendet wird
 2. Handy interne xml config beim ersten starten
- Proxy soll suchanfragen, download anfragen, etc mitloggen. Ob ips der handys mitgeloggt werden soll oder nicht, entscheidet letztendlich der investor)

Es gibt nur verschlüsselte Verbindungen:

- Handy <-> Proxy: Mittels Crypto Library(zb. <https://github.com/rtyley/spongycastle>) 256 bit AES (Proxy hat 256bit Cryptokey als config parameter gespeichert, leicht änderbar; handy hat 256 bit aes crypt key in xml config gespeichert)
- Proxy<->Content Server: Nutze HTTPS Schnittstelle des Content servers
- Handy<-> Content Server: Nutze HTTPS Schnittstelle des Content servers, ist https nicht verfügbar nutze http

Beschreibung des Proxys:

Der Proxy ist ein in Java geschriebenes Tool, welches als Vermittlungsstelle zwischen Content Server und Handy ist. Das Handy verbindet sich zu einem Proxy mittels DOMAIN oder IP (Handy hat diese Daten als xml config gespeichert) zu einem definierenden Proxy TCP Port. Valide Proxy Ports beginnen ab 1025. Die Proxy Software muss als Linux benutzer Root und als normalpriorer Benutzer vollständig laufen.

Definition:

- Proxy IP: 1.1.1.1
- Content Server IP: 3.3.3.3 (der Entwicklungsserver ist www.buzzerstar.com)

Verbindungsablauf für einen Download/Stream vom Handy:

1. Handy stellt Verbindung zum Proxy(1.1.1.1.) her (mittels AES256bit encryption)
2. Handy stellt einen Suchbegriff an den Proxy
3. Proxy nimmt suchbegriff, logt diverse Statistiken mit und stell einen Suchrequest via HTTPS an urlencodded die Content Server IP
<https://3.3.3.3/search.php?q=suchbegriff> (es gibt das Feature nicht nach pornos zu suchen: <https://3.3.3.3/search.php?q=suchbegriff&noporn=1>)

4. der proxy soll verschiedene content server gleichzeitig anfragen und verarbeiten können
5. Proxy bekommt vom Content Server eine liste mit ergebnisse zurück, im xml protocol format
6. Proxy stösst nebenbei eine Youtube Suche an (anzahl ist flexibel konfigurierbar, geordnet nach relevanz und aktualität – Youtube api von google benutzen)
7. Proxy sortiert die ergebnisse vom content server und die youtube ergebnisse, sodass
 - a. Zuerst KINOFILME
 - b. Dann Serien
 - c. Dann youtube (youtube icon nehmen für youtube videos)
 - d. Dann anime/comic
 - e. Dann porn
8. Proxy liefert alle ergebnisse aus: handy stellt dar:
 - a. Max 30 ergebnisse pro screen (xml config), stell man weniger ergebnisse ein, verändern sie die relationen der kinofilm und serien gegenüber youtube,anime,porn nicht), dann muss man runterwischen um den nächsten inhaltsscreen zu bekommen. Beim ersten mal suchen muss Text: „Bitte scrolle weiter runter, um neue Ergebnisse zu deine Suchanfrage zu bekommen“. stehen (xml config)
 - b. 9 x Kinofilme
 - c. 9 x Serien
 - d. 6 x youtube
 - e. 3 x anime
 - f. 3 x porn
9. Handyuser wählt Stream oder Download aus
10. Download / Stream startet (via HTTPS wenn möglich, sonst http vom content server)

Verbindungsablauf für einen News Screen Request:

1. Handy verbindet sich zum Proxy (AES256bit)
2. Handy sagt dem Proxy (mittels standard xml protocoll): Liefere mir den aktuellen News screen aus
3. Proxy liest news screen xml ein und liefert diesen an das handy aus
4. Handy parst xml (xml fehler werden ignoriert) und stellt die news da
5. Handy speichert die news für N-Tage (xml config) und stellt N-Tage keinen neuen News request

Verbindungsaufbau für einen Youtube Download Request

1. Handy zu proxy verbindung
2. Handy sagt proxy(mittels standard xml protocol: ich möchte <http://www.youtube.com/watch?v=0SeC81tEazY> downloaden

3. Proxy nutzt externe JAVA bibliothek (muss im quellcode vorliegen, zwecks anpassbarkeit) um youtube video zu downloaden – Qualität mindestens MP4 format
4. Proxy: gedownloadete youtube datei wird in einen cryptischen ordern (random md5 wert bilden) verschoben, der via lighttpd zugreifbar ist
 - a. Sprich: auf dem Proxy server läuft ein lighttpd (webserver)
 - b. Proxy lädt datei und schiebt sie in ein MD5 verzeichnis vom Webserver
 - c. Proxy speichert sich die webadresse (=URI) des gedownloadeten videos
5. Proxy gibt HTTPS oder http link via xml standard kommunikationsprotokoll an das Handy zurück
6. Handy kann youtube datei vom proxy automatisch laden und stellt den download in der history da (incl youtube icon)
7. Proxy logt alles mit
8. Während der Proxy die datei lädt: Handy zeigt an, „wir bereiten deinen Youtube Download vor. Er wird automatisch heruntergeladen und du kannst die BuzzerStar App weiter benutzen.“ (xml sprachfile, config file)

Allgemeines um Youtube Support:

- Suche erfolgt im Proxy (Youtube Suche über Kanäle, Videos und Wiedergabelisten)
 - Als Ergebniss der Youtube suche werden nur gültige Youtube links ans handy weitergeleitet; keine Playlist oder sonstige youtube urls!!
- Separate youtube config für den proxy (anzahl ergebnisse, api keys, aktiv/inaktiv, etc)
- Es müssen mehrere Youtube Api Keys hinterlegt werden können:
 - Existieren mehrere: zufallsauswahl welcher genommen wird
 - Einer: nimm diesen
 - Melde, (proxy logfile) wenn ein Api Key fehlerhaft/deaktiviert wurde
 - Überspringe deaktivierte Api Keys
 - Speichere pro Api Key die bereits ausgelieferte Anzahl von Requests pro Tag
 - Wenn nur 1000 erlaubt werden, nimm ab 1001 einen neuen
- Youtube ergebnisse im handy mit Youtube Icon (unten) markieren
- Handy App muss reguläre Youtube Video Links unterstützen (Beispiel: <https://www.youtube.com/watch?v=5-405Vvn3OU>) als Stream (eventuell Youtube Api zum darstellen von Youtube Screens verwenden, zum Download extra library auf dem Proxy nutzen)
- Welche Qualität das Youtube video hat wird in der youtube proxy configuration festgelegt
- Soweit wie möglich aktuellste, offizielle YouTube Api zum suchen benutzen, Logik auf Proxy auslagern
- Wenn Youtube Api Key auf dem Handy gebraucht wird, muss es via Proxy XML Protokoll gesendet werden (proxy muss datenbank mit validen api keys vorhalten)
- Werbung von Youtube muss ignoriert bzw automatisch übersprungen werden
- Youtube ICON:

You  **Tube**