

Mobile Devices User Protocol Version 1.0

Stand: 07.11.2005

Abkürzungsverzeichnis

API	Application Programming Interface
CP	Content Provider
GPRS	Global Packet Radio Standard
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
ID	Identificator
IMEI	International Mobile Equipment Identifier
IMSI	International Mobile Subscriber Identifier
IP	Internet Protokoll
ISO	International Stardardization Organisation
ITU	International Telecommunication Union
LBS	Location based service
MDPP	Mobile Devices Payment Protocol
MSISDN	Mobile Number Integrated Services Digital Number
MU	Mobile User
NO	Network Operator
PDA	Personell digital assistant
RADIUS	Remote Authentication Dial In User Service
SIM	Subscriber Identification Module
SOAP	Simple Object Access Protocol
SSL	Secure Socket Layer
TLS	Transport Layer Security
UMTS	Universal Mobile Telecommunication Standard
URL	Unified Resource Locator

Inhaltsverzeichnis

Abkürzungsverzeichnis.....	2
Inhaltsverzeichnis	3
Abbildungsverzeichnis	3
Copyright	3
Terminologie.....	4
Technische Beschreibung	6
UserData Anfrage.....	8
Mobile User Authentisierung.....	9
UserData Bereitstellungsprozess	9
„trusted Token“ Validierung durch den Content Provider.....	12
RSA Schlüsselaustausch.....	13
Beispiel: Erzeugen einer Signatur.....	14
Beispiel: Verifizieren einer Signatur	14
Sicherheitsbetrachtungen	15
Man-in-the-Middle	15
Replay Angriffe	15
Reproduktion von Tokens	15
Identifikation des Benutzers	15
Betrug durch Parameter Manipulation.....	15
Betrug durch falsche Token Parameter.....	15
Generelle Betrachtung	15
Referenz.....	16
Annex 1 Parameter Definition.....	17

Abbildungsverzeichnis

Abbildung 1: Übersicht.....	7
Abbildung 2: Kommunikationsbeziehungen.....	7

Copyright

Dieses Dokument sowie sein Inhalt unterliegen dem Copyright der Cyber-Dynamix GmbH, Nürnberg © 2005. Jegliche Nutzungen, Veröffentlichungen auf elektronischer, schriftlicher oder anderer Weise sowie Vervielfältigungen bzw. Kopien sind ohne ausdrücklicher schriftlicher Genehmigung untersagt. Der Herausgeber und der Autor behalten sich alle Rechte vor.

Terminologie

Content Provider

Anbieter von Internet Inhalten, z.B.: Nachrichten, Klingeltöne, Videos, Animationen, Spielen, etc.

Mobile User

Nutzer eines mobilen Endgerätes [Device] mit Zugang zu und innerhalb von einem Mobilfunknetz eines Mobilfunknetzbetreibers [Network Operator] und Konsument von Mobilfunk- und Internet-basierten Angeboten auf Basis von vorausbezahlten Guthaben (PrePaid) oder Verträgen (PostPaid) mit zyklischer Rechnungsstellung.

Network Operator

Betreiber eines Mobilfunknetzes und der dazugehörigen Infrastruktur zur Authentisierung, Autorisierung und Abrechnung von Kunden und Mobilfunkdienstleistungen sowie zur Bereitstellung von Zusatzleistungen, z.B. Geopositionsdaten, Messaging Dienstleistungen, Short- und MultiMedia Diensten, etc.

Content

Kostenfreie und –pflichtige Informationen oder Dienstleistungen eines Content Providers.

Token

Definiertes und zusammengehöriges Set an Informationen.

Signatur

Digitale nicht reproduzierbare Kennzeichnung eines Tokens unter Einsatz von kryptographischen „public key“ Verfahren.

MSISDN

Eindeutige Mobilfunknummer zur Identifikation eines MU innerhalb eines Mobilfunknetzes.

RSA Kryptverfahren

Asymmetrisches Verschlüsselungsverfahren der Mathematiker Ronald L. Rivest, Adi Shamir und Leonard Adleman.

Autorisierung

Ermittlung der Berechtigung

Authentisierung

Ermittlung der Echtheit

Autorisierungsobjekt

Server basierte Anwendung des NO zur Autorisierung, persistenten Speicherung, Guthabenprüfung, Abrechnung und Beantwortung von MU Autorisierungsanfragen.

Device

Mobiles Endgerät mit SIM Karte (z.B. Mobiltelefon) oder Endgerät (z.B. PDA, Notebook, ...) mit Zugang zu einem Mobiletelefon(z.B. über Infrarot-, Kabel- oder Bluetooth Schnittstelle)

Technische Beschreibung

Das MDU-Protokoll wurde zur Bereitstellung von Kundendaten für Content Anbieter, wie zum Beispiel Informationen über das Access Netzwerk (UMTS, GPRS, WLAN, ...), die zur Verfügung stehende Bandbreite, Standortdaten oder MSISDN, in mobilen Datennetzen entwickelt. Es beruht auf dem Prinzip der dezentralen Informationsbereitstellung, zentralen Benutzerauthentisierung und signierten Nachrichten [Token], welche zwischen einem Content Anbieter [Content Provider], einem Konsumenten [Mobile User] und einem Mobilfunkbetreiber [Network Operator] ausgetauscht werden. Das MDU-Protokoll ist erweiterbar, so dass NO spezifische User Daten, z.B. Nicknames, eMail Adressen, ... ebenfalls bereitgestellt werden können.

Die Kommunikation zwischen einem CP und einem MU erfolgt über HTTP (RFC2616), wobei die Informationsübermittlung direkt als Parameter in HTML bzw. als http Query String erfolgt. Dieser Mechanismus ist im folgenden Text detailliert beschrieben.

Die Kommunikation zwischen einem MU und einem NO erfolgt entweder über HTTP (siehe oben) oder als SOAP basierter Webservice Aufruf, welcher entweder direkt aus einer HTML-Seite, über eine API oder eine Applikation erfolgen kann.

Weiterhin kommen „trusted Token“ zur Autorisierung von Anfragen der Content Provider zur Anwendung, welche durch die authentifizierende Instanz (NO) digital signiert werden und dem CP somit die Möglichkeit bieten, zusätzliche Informationen über den Kunden zu erlangen.

Das MDU-Protokoll erfordert prinzipiell keine weitere Online Kommunikation zwischen Content Providern und Network Operatoren. Ebenso werden keine zusätzlichen Kommunikationskomponenten, wie z.B. gateways oder access proxies benötigt.

Die nachfolgende Graphik verdeutlicht die Kommunikationsbeziehungen:

Ein MU kommuniziert mit einem Content Provider über das Internet um Informationen über einen MU zu erhalten. Hierbei wird der Zugang zum Internet über das Mobilfunknetz eines Network Operators gewährleistet.

Zum Erlangen von Benutzerinformationen kommuniziert der Content Provider über den Mobile User mit dem Network Operator.

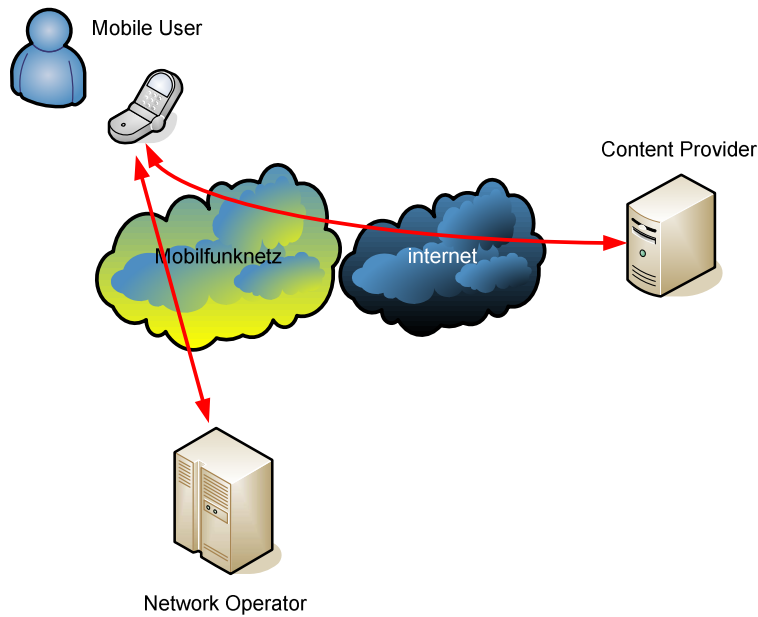


Abbildung 1: Übersicht

Das folgende Flussdiagramm zeigt die Interaktionen zwischen MU, CP und NO. Die Details der Nachrichten sind in den folgenden Kapiteln beschrieben.

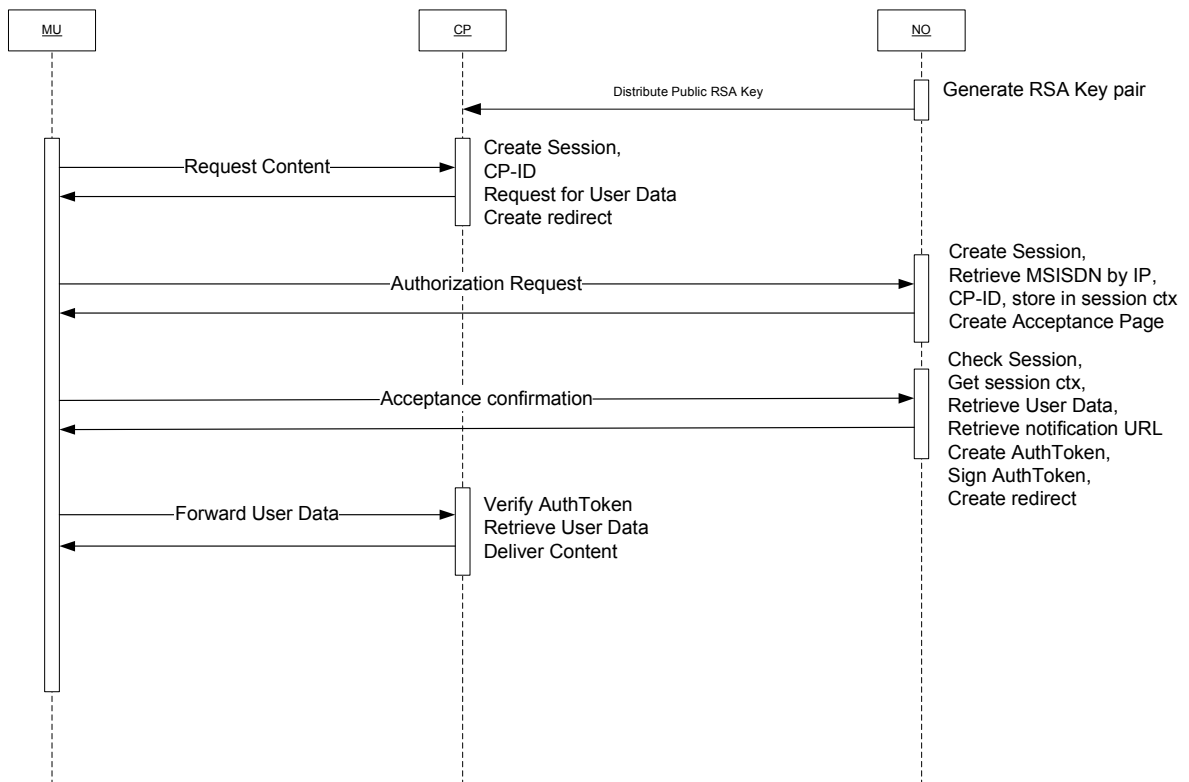


Abbildung 2: Kommunikationsbeziehungen

UserData Anfrage

Sobald ein Content Provider für die Erbringung seiner Dienstleistung weiterführende Informationen über einen MU benötigt, wird er diese mittels einer UserData Anfrage vom NO anfordern. Die Anforderung erfolgt über die Definition eines Parametersatzes innerhalb des URL Query-Strings, welche mittels eines http Redirects auf das UserData-Objekt (URL) des Network Operators zeigt.

Hierzu erzeugt der Content Provider den folgenden http Response Code:

```
HTTP/1.1 301 Moved Permanently
Date: Wed, 7 Nov 2005 07:31:26 GMT
Server: CP-host
Location: http://NO-host/UserData-
Object?SessionID=98846AF87897B0AC6876868F376&CPID=
Anbieter+123&Request=MSISDN;Bearer;LBS
Content-Length: 0
Content-Type: text/html
Connection: close
```

Die Parameter besitzen folgende Bedeutung:

CPID	Eindeutiger Identifikationscode des Content Providers. Die CPID muss vom Network Operator vergeben werden.
SessionID	Session Identifikator des Content Providers
Request	Angefragte Benutzerinformation; die angeforderten Daten werden durch Kommata separiert

Der Parametersatz muss vom CP im Session Context / Datenbank gespeichert werden, da er für die spätere Validierung der Signatur und damit zur Überprüfung der Autorisationsantwort erforderlich ist.

Sobald der Browser des mobilen Devices den http Redirect auflöst, wird ein http request gegen das UserData-Objekt des Network Operators mit den oben definierten Parametern gestellt.

```
GET http://NO-host/UserData-
Object?SessionID=98846AF87897B0AC6876868F376&CPID=
Anbieter+123&Request=MSISDN;Bearer;LBS HTTP/1.1
Host: NO-host
User-Agent: xyz
Accept: */*
```

Mobile User Authentisierung

Der NO muss vor jeder Auslieferung benutzerbezogener Daten die Authentizität des Mobile Users überprüfen. Dies geschieht anhand der Client IP Adresse mittels einer Abfrage der RADIUS Datenbank, in welcher die MSISDN des MU und seine derzeit zugewiesene IP Adresse hinterlegt sind.

Das UserData-Objekt prüft – nach erfolgreicher Authentisierung des MU – die übermittelten Parameter auf Plausibilität und Gültigkeit und hält diese für eine spätere Verarbeitung im Session Context vor. Die übermittelten Informationen (Query String) werden gegen ein Repository abgeglichen. Weichen die Informationen von den bei NO hinterlegten ab (z.B. CP-ID unbekannt), so wird der Vorgang mit einer Fehlermeldung an den MU abgebrochen.

Fällt das Ergebnis der Überprüfung positiv aus, so wird anhand der übermittelten Parameter eine HTML Seite generiert, welche den MU über die Anforderung des CP informiert und das Einverständnis zur Übermittlung einholt.

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>Information</title>
</head>
<body>
Der Diensteanbieter <Anbieter 123> hat um die Bereitstellung folgender
Informationen gebeten:
<p>
Ihre Mobilfunknummer
</p>
<form method="POST" action=http://NO-host/UserData-Object accept-
charset="ISO8859-1">
  <input type="submit" value="Übermitteln">
  <...>
</form>
</body>
</html>
```

Gestattet der MU die Übermittlung der angeforderten Daten durch Betätigung des "Submit-Buttons", so wird der Bereitstellungsprozess beim NO angestoßen.

UserData Bereitstellungsprozess

Nachdem der MU der Übermittlung der angeforderten Daten zugestimmt hat, werden die zu übermittelnden User Daten durch das UserData-Objekt ermittelt. Im Anschluß wird durch das UserData-Objekt eine digitale Signatur erzeugt und zusammen mit den im Session Context vorgehaltenen Parametern, den ermittelten User Daten sowie der MSISDN des MU als Datensatz persistent beim NO gespeichert.

Die CP Session Information und die Signatur der CP Parameter sowie die angeforderten User Daten werden innerhalb des vom Bereitstellungsobjektes erzeugten http redirects an den MU zurückgegeben. Hierbei wird als Ziel URL die beim NO gespeicherte notification URL herangezogen.

Hierzu erzeugt der NO den folgenden http Response Code:

```
HTTP/1.1 301 Moved Permanently
Date: Wed, 7 Nov 2005 07:31:26 GMT
Server: NO-host
Location: http://CP-host/notification-Object?Token=
b5IwJtb30YLD8j6PUHHGni3VjVgYUv4XvnVP6TIhpqc+f4DdLLUr8CfY8uUeVABx2napNrt/S+N
DWbfTXOAIWZUtGckaiCm4L1hjvvr21CjCcJfLxcn3Cg4umfWR6CrpuFoiedyOHGVaruJxMhUj9
W2kd5/9bLnSLWPYXX77usU9vFdOJYKr+uc0DdKYWbaSvzLYH5uWOK9ddjm+k2ukVv4hIOHHdbTD
PIjE+h2w3cN2DI2iyM/9IU+JD8ofREphOqM3sD+hHcwCYQs3cuPemH/TnH0v5hC0CfwJ3SnMFfE
gqIdgIQlyjQwOP7m5CQNL/uHUaHoeGcmiQ1gD1PzFA==&OperatorID=XYZ+Mobilfunk+GmbH&
MSISDN=01721234567&Bearer=GPRS&LBS=23,000|58,021|120
Content-Length: 0
Content-Type: text/html
Connection: close
```

Hierbei bedeutet:

SessionID	Die ursprüngliche SessionID des Content Providers
Token	„trusted Token“, digitale Signatur
OperatorID	Der Name des autorisierenden Mobilfunk Betreibers.
MSISDN	Rufnummer des MU
Bearer	Access Netzwerk Identifikation: <ul style="list-style-type: none"> • GPRS • UMTS • WLAN • DSL • Internet
LBS	Geo Positionsdaten in X-Pos Y-Pos Radius

Die in der initialen Autorisierungsanfrage mitgegebenen Parameter werden digital signiert:

Diese sind beispielsweise (vergleiche oben):

```
SessionID%3D98846AF87897B0AC6876868F376%26CPID%3D+Anbieter%2B123%26Request%
3DMSISDN%3BBearer%3BLBS
```

Die Signatur wird durch den nicht-öffentlichen RSA Key des NO erzeugt [siehe Kapitel RSA Schlüsselaustausch]. Das Ergebnis ist in base64 zu codieren:

```
b5IwJtb30YLD8j6PUHHGni3VjVgYUv4XvnVP6TIhpqc+f4DdLLUr8CfY8uUeVABx2napNrt/S+N
DWbfTXOAIWZUtGckaiCm4L1hjvvr21CjCcJfLxcn3Cg4umfWR6CrpuFoiedyOHGVaruJxMhUj9
W2kd5/9bLnSLWPYXX77usU9vFdOJYKr+uc0DdKYWbaSvzLYH5uWOK9ddjm+k2ukVv4hIOHHdbTD
PIjE+h2w3cN2DI2iyM/9IU+JD8ofREphOqM3sD+hHcwCYQs3cuPemH/TnH0v5hC0CfwJ3SnMFfE
gqIdgIQlyjQwOP7m5CQNL/uHUaHoeGcmiQ1gD1PzFA==
```

Die Signatur inkludiert den Hash-Wert der Übergabeparameter. Diese können zusammen mit dem öffentlichen Schlüssel des NO durch den CP validiert werden.

Innerhalb der Bestätigungsantwort werden die CP Session Information und die Signatur – jetzt als „trusted Token bezeichnet“ – die OperatorID und die User Daten als URL Query String Parameter dargestellt. Die Request hat die beim NO registrierte CP notification URL zum Ziel.

Sobald der Browser des mobilen Devices den http Redirect auflöst, wird ein http request gegen das Notification-Objekt des Content Providers mit den oben definierten Parametern gestellt.

```
GET http://CP-host/notification-Object?Token=
b5IwJtb30YLD8j6PUHHGni3VjVgYUv4XvnVP6TIhpqc+f4DdLLUr8CfY8uUeVABx2napNrt/S+N
DWbfTXOAIWZUtGckaiCm4Llhjvvr21CjCcJFLxcn3Cg4umfWR6CrpuFoiedyOHGVaruJxMhUj9
W2kd5/9bLnSLWPYXX77usU9vFdOJYKr+uc0DdKYWbaSvzLYH5uWOK9ddjm+k2ukVv4hIOHHdbTD
PIjE+h2w3cN2DI2iyM/9IU+JD8ofREphOqM3sD+hHcwCYQs3cuPemH/TnH0v5hC0CfwJ3SnMFfE
gqIdgIQlyjQwOP7m5CQNL/uHUaHoeGcmiQ1gDlPzFA==&OperatorID=XYZ+Mobilfunk+GmbH&
MSISDN=01721234567&Bearer=GPRS&LBS=23,000|58,021|120 HTTP/1.1
Host: CP-host
Content-Type: text/html
Connection: close
```

Die Übertragung der Parameter erfolgt URL-encoded.

„trusted Token“ Validierung durch den Content Provider

Der Content Provider erhält nun eine http Anfrage welche eine SessionID und einen „trusted Token“ als URL Parameter beinhaltet.

Anhand der SessionID kann der CP die ursprünglichen Parameter der initialen MU Anfrage aus dem Context der mit der SessionID assoziierten Session ermittelt.

Diese sind in diesem Beispiel:

```
SessionID%3D98846AF87897B0AC6876868F376%26CPID%3D+Anbieter%2B123%26Request%3DMSISDN%3BBearer%3BLBS
```

Mithilfe des als Parameter enthaltenen „trusted Tokens“ kann der CP nun die Authentizität und somit die Gültigkeit der Anfrage ermitteln.

Hierbei wird der „trusted Token“ – die digitale Signatur – zusammen mit den ursprünglichen Parametern gegen der öffentlichen Schlüssel des NO validiert.

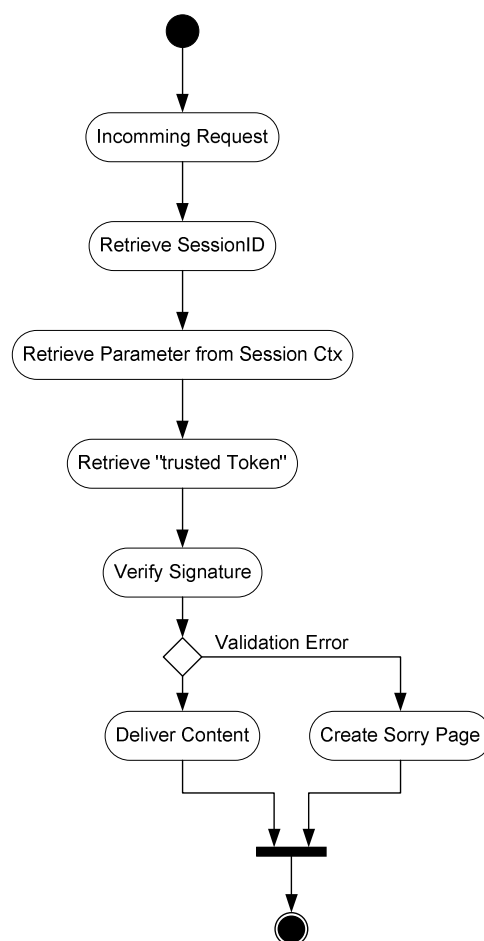


Abbildung 3: „trusted Token“ Validierung

Nach erfolgreicher Auslieferung des Contents an den MU wird der Session Context durch den CP zerstört und ggf. eine neue Session instanziiert.

RSA Schlüsselaustausch

Bevor das MDUP zum Einsatz kommen kann, muss ein NO ein RSA Schlüsselpaar generieren. Die Länge des zu erzeugenden Schlüssels bestimmt die Sicherheit des Verfahrens und sollte nicht unter 2048 Bit liegen.

Nach erfolgreicher Generierung eines Schlüsselpaares, ist der öffentliche und der private Schlüssel zu trennen und separate zu verwahren. Der öffentliche Schlüssel ist den Content Providern auszuhändigen.

Das folgende Beispiel zeigt die Schlüsselgenerierung mittels der freien Software openssl:

```
$ openssl
OpenSSL> version
OpenSSL 0.9.7b 10 Apr 2003
OpenSSL> genrsa -out PrivateKey 2048
Loading 'screen' into random state - done
Generating RSA private key, 2048 bit long modulus
.....+++
.....+++
e is 65537 (0x10001)
OpenSSL> rsa -in PrivateKey -out PublicKey -pubout
writing RSA key
```

In diesem Beispiel wird ein neues Schlüsselpaar mit einer Länge von 2048 Bit generiert und gemeinsam in der Datei `PrivateKey` abgelegt. Im Anschluß wird der öffentliche Schlüssel extrahiert und in die Datei `PublicKey` geschrieben.

Beispiel: Nicht-öffentlicher Schlüssel aus `PrivateKey`

```
-----BEGIN RSA PRIVATE KEY-----
MIIEpQIBAAKCAQEAAzmhp6m4dfZqHhyI196Poss7HGgxswJx4KuoopZktrZv14eaY
NSmMe4GBuY9bOfuRjnJK3y+Uno jFcSKS9RGvbaTormXR3tMhKb9DeDDYsK+grcQS
N4J4jInS8IFl35IXHQdjn+VmIN3eWy3Y/NO/pSwDxYNWp4B7Qvi8zckvTjnH8MSz
5j0zJn9DE70pvAXwfYvb3tgwzeEeaNwpxpqvDVdaEUyBmmNyuQipn8aV6AcWYWWP
VxL6Ikml3l7iIUrr+KNK43djWbvbNj/p1EhMeT1sTOyUDZ8JzNbuY5M9BdkjpXGWF
GX7sebEcVgXOQVuEPRSxQdANzena/aOn3CezewIDAQABAoIBAQCAZfn9IYK236nv
JhjsM8pNcNpQIFCGgM8/4unmGaOQm7QZEpkcePwq2CUaiVL2fsfNGNcafz60X+By
wE8qpBW83xjfy/dp5Wx7y77sxXYGczpA3fd3EHJs+k99zQDSUs6OAEq47LrbNylb
DuosXIya8wZl144QJJyIzLkmUMBcp/XsE7IJ9aBRsIUTh6oua0afxYRzhIESi+ii
J6r23VOEp/s/JU6SnuYiWLUqMtOaAQeKlrV4ZjPULvSIyWlpXN8B1hVg4lPBYWQC
/SIRxCVCpYOf8L+Ztur7DD2aVJz2zPJobWDckDRnJBOH76hPelrM3lAZ+Ohmzsoe
ALHcQ8wBAoGBAPqCEQtPwvaI/7TkD2plpF9Tom9NCHFdDWzCLLSAR15g051HtBIW
eth5hf6GKEXKYU5Gzo84NWREecLI+iW62vAyc3rTlOB55Tn+4rCNatpewUuF3xs7
aint4jki0vZGhREoFQepZ923+zvvXKo00SI40TPyIkmkMJLq/hpptgy5AoGBANLu
16OUOA/1zpVlDJs66xSCpuLD/IQ4FG1xiMt87vmn6YCpNwYS8eTXByRzY6bXlpxq
xtMDw/0vlAVWMQI5ZJ+nRPlf6zc9edTYa4dWmkF8Y3yFhZu/5VfoDhEl8LGYZL2/
KqeSvvBEkiRx5LgYmjG/5hIOhft5WiE18c+Sz2/TAoGBALhgewMEVc72zp3pLz91
6CFxgSDCsnv9rR/bWuQPdbuIwNfmKpcVjJ0/9Gt9eq7DYhMm8mlfSYzfCW9gVRzo
BrS7rVs911nQ3fJts5OWwoqvKz3W7nsw09bwi4zaIMO6673Q7omRGi2KeJOfx99
IFg70V6WXL4u5sF7zELMg32hAoGBALS0W1bPNwwtSFLiY99kpVpH59LjliRrqsxr
9IZnvI9zE27fCL2SY1rqADtxA1E+5zuBeF30nuX76bJ8ubWvF45TDZsaWndTmlkH
I2+peLNbbhuSg/j/d+S64To9p6tt4/hOmqs+44cRJ6ZDUG+K3CZ8wQx9FVUMbHOB
NGzg2AdzAoGAGuCCoUp314ux5SK8Dqiv5aLqJOLXt6THCPcXk9YMHhprgVbIM+aY
```

```
8CaI3x/Yyrvb7TeY9HTsAgKn+/fmBB8Pwt++K+XzpUOGR8qjqfx6Y5xsS7PZZ3Mx
SKfufb8bPEZKKb/uGPMDawGs73GBD7HmSFCn3DPBVi5KQdopQDwBQ=
-----END RSA PRIVATE KEY-----
```

Beispiel: Öffentlicher Schlüssel aus `PublicKey`

```
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAzmhp6m4dfZgHhyI196Po
ss7HGgxswJx4KuoopZktrZv14eaYNSmMe4GBuY9bOfuRjnJK3y+UnojFcSKS9RGv
baTormXR3tMhKb9DeDDYsK+grcQSN4J4jInS8IF135IXHQdjn+VmIN3eWy3Y/NO/
pSwDxYNWp4B7Qvi8zckvTjnH8MSz5j0zJn9DE70pvAXwfYvb3tgwzeEeaNwpxpqv
DVdaEUYBmmNyuQipn8aV6AcWYWwPVxL6Ikm3l7iIUrr+KNK43djWbvbnj/plEhMe
T1sTOyUDZ8JzNbuY5M9BdkjpXGWF7sebEcVgXOQVuEprSxQdANzena/aOn3Cez
ewIDAQAB
-----END PUBLIC KEY-----
```

Beispiel: Erzeugen einer Signatur

Die zu signierenden Parameter sind in der Datei `Parameterblock` gespeichert.

```
$ openssl dgst -sign PrivateKey -out Token < Parameterblock
```

Die Signatur ist nach diesem Aufruf in der Datei `Token` abgelegt.

Beispiel: Verifizieren einer Signatur

Die zu verifizierende Signatur ist in der Datei `Token` und die Parameter in der Datei `Parameterblock` abgelegt.

```
$ openssl dgst -verify PublicKey -signature Token < Parameterblock
Verification OK
```

Sicherheitsbetrachtungen

Man-in-the-Middle

Durch die Nutzung von End2End SSL/TLS Verbindungen kann eine Man-in-the-Middle Attacke wirkungsvoll unterbunden werden.

Replay Angriffe

Durch die temporäre Erzeugung des CP SessionID und die Kopplung der Zugangsinformationen an die SessionID sind Replay Attacken praktisch ausgeschlossen, da nach der erfolgreichen Auslieferung des Contents die Session zerstört wird.

Reproduktion von Tokens

Durch die Verwendung des asynchronen RSA Verfahrens ist dies nur möglich, wenn ein potentieller Angreifer in Besitz des nicht-öffentlichen Schlüssels gelangt. Dieser ist jedoch, wie prinzipiell alle Geheimnisse – für Dritte unzugänglich aufzubewahren.

Identifikation des Benutzers

Eine Identifikation des Benutzers seitens des Content Providers ist nicht möglich, da keinerlei personenbezogenen Informationen, wie z.B. MSISDN oder wiederkehrende Informationen, z.B. XID übertragen werden.

Betrug durch Parameter Manipulation

Durch die erforderliche Registrierung des Content Providers beim Network Operator und die dadurch mögliche Plausibilitätsprüfung der übermittelten Daten, können Manipulationen erkannt und der Autorisationsprozess abgebrochen werden. Ein CP erkennt maximal die Zugehörigkeit eines MU zu einem NO.

Betrug durch falsche Token Parameter

Bedingt durch die Tatsache, dass die Content relevanten Informationen seitens des CP vorgehalten werden und nicht an den zu übermittelten „trusted Token“ geknüpft sind, kann ein CP Manipulationen erkennen und die Content Auslieferung verhindern.

Generelle Betrachtung

Durch die dezentrale Datenhaltung und die Möglichkeit übertragene Parameter abzugleichen, fallen Manipulationen auf. Weiterhin gewährleistet das RSA Signatur Verfahren eine zuverlässige Verifikation von Autorisierungstoken.

Weiterhin sind Angriffe gegen das Autorisierungssystem des NO als unwahrscheinlich einzustufen, da es nur innerhalb des GPRS/UMTS –Netzwerkes eines NO erreichbar ist und keinen Zugang aus dem Internet besitzt.

Betrügerische Manipulationen gegenüber dem Autorisierungssystem sind immer an eine SIM Karte gebunden. Somit können böswillige Kunden erkannt und blockiert werden. Bei der missbräuchlichen Nutzung von z.B. entwendeten SIM Karten besteht die Möglichkeit der netzseitigen Sperrung.

Referenz

RFC2616	Hypertext Transfer Protocol HTTP/1.1
RFC2138	Remote Authentication Dial In User Service
RFC2313	PKCS #1 RSA Encryption Version 1.5
RFC2246	Transport Layer Security
RFC1738	URL encoding
HTML	W3C-Recommendation 24. December 1999
openssl	open source software, siehe www.openssl.org
ISO 8601	International Date and Time notation
ITU-T E.164	International Public Telecommunication Numbering Plan

Annex 1 Parameter Definition

Name	Typ	Länge	Inhalt	Kommentar
CPID	String	16	Zwischen NO & CP vereinbarter Service Provider Identifikator	
SessionID	String	64	Session Identifikator eines Content Providers	
OperatorID	String	64	Kennung eines Network Operators	
MSISDN	String	12	Mobile Rufnummer des MU im ITU-T E.164Format	
Bearer	String	20	Access Netzwerk Informationen	GPRS UMTS_64 UMTS_128 UMTS_384 WLAN DSL_1 DSL_2 DSL_3
LBS	String	20	Geo Positionsdaten X-Pos Y-Pos Radius der Zelle	
Token	String	256	Digitale Signatur	