Abstract

The explosion of new types of end-user devices, as well as the increase in the number of users with broadband connection in private households, opens up the opportunity to provide access to documents and services residing in the home domain from remote locations. Also, the increased mobility of users, as well as the need to be able to work in all locations, further adds to the requirement of remote document access. The contribution of this paper is threefold. First, it introduces the Mobile Home Access system, which simplifies remote access to resources in a home domain. The Mobile Home Access (MHA) system is based on the Service-Oriented Computing concept and realized using XML Web Services for communication between the home domain and the end-user device, while employing the Common Internet File System (CIFS) network file system to allow appropriate file access in the home domain. The second contribution is the mechanism whereby such a service can be deployed when network elements like firewalls and NAT routers are introduced. The challenge of establishing a connection from a remote device is general for both client–server services and peer-to-peer services. The paper proposes and describes a design and implementation of a solution using XML Web Services technologies, and thus shows how NAT traversal technology can be integrated with such middleware. The third contribution of this paper is a performance analysis of the current implementation of the Mobile Home Access service. The results of the analysis should be interesting for the applicability of XML Web Services in mobile computing in general.

Keywords: Mobile middleware; XML Web Services; Personal services; Remote file access; Mobile services; Mobile devices; VPN

1. Introduction

New types of end-user devices are arriving in the market all the time, e.g., mp3-players, digital cameras, PDAs, etc. The services provided by these devices are mostly based around some type of content, e.g., mp3-files, photos, text documents, etc. Also, devices that combine and specialise two or more functions are emerging, e.g., cellular phones that are improved for playing digital music (e.g., the Sony Ericsson W800i and Nokia N91) or with improved digital camera functionality (e.g., the Nokia N90). The latter devices are focused around two primary concepts; content (music/photos) and communication (voice/data). Increasingly, private households have one or several computers connected together in a Local Area Network (LAN), and these are quite often connected to the Internet through a broadband connection.
Combining these facts, it is quite natural to anticipate the requirement that end-users should have remote access to documents and services residing on the computers in the home network, also when being mobile. If the home network is open, i.e., unprotected by firewalls, the access can be enabled quite simply by using existing software solutions for some devices like laptops, although access from mobile phones is still not trivial.

However, the situation with open home networks is quite unlikely with all the fraudulent intrusions that happen nowadays. Moreover, the home network may not receive a global IP address but only a local one because the broadband operator may only have a limited number of IP addresses and thus uses Network Address Translation (NAT) to cater for several computers on the home network.

This paper proposes a new solution enabling mobile access to the home network from various devices, based on the Service-Oriented Computing (SOC) [1] concept and realized using XML Web Services. This brings improvements with respect to reduced complexity, less resource consumption and a more user-friendly solution, compared to existing solutions like Virtual Private Networks (VPN). Mobile Home Access is designed to be a complete solution, end-to-end, which enables flexible remote file access from eventually many types of mobile devices.

The paper starts by a thorough analysis of existing solutions in Section 2. Section 3 provides an overview of XML Web Services, and in particular the parts that are relevant to the proposed solution. Section 4 describes the problem domain; the home domain/home network, and common characteristics of these. Finally, Section 5 introduces the Mobile Home Access solution and gives some details of the system. Section 6 provides a performance evaluation of the proposed solution, and Section 7 discusses the security issues surrounding the solution. The paper ends with a description of future works and a conclusion.

2. Background

2.1. Virtual private networks (VPN)

Currently, VPNs (e.g., based on IPSEC [2]) are used to allow access to file systems in remote locations from, e.g., laptop computers by users that are traveling (often referred to as a road warrior configuration) or to access corporate networks from home. One of the advantages of using this solution is that it provides a secure tunnel between two locations, across the public Internet. Strong cryptography is used both for authentication and authorization of users as well as for encryption of content that travels across the tunnel. Also, as suggested by the name, when using VPN the networks connected by the tunnel will appear as one network to the user, thereby allowing easy access to well-known features (e.g., Windows’ network neighbourhood) of geographically separated networks. The technology is often used for interconnecting branch offices of enterprises. It can, of course, be used for Mobile Home Access. Unfortunately, this solution has a couple of drawbacks which makes it unsuitable for the scenarios depicted in the introduction.

2.1.1. Complexity

VPN solutions require both a client on the mobile device and a server in the user home network. The server, in particular, is often complex to install and configure since in-depth knowledge of computer and networking technology is often required. This means that ordinary users cannot do this task themselves. In addition to the installation of the VPN server itself, such solutions require advanced configuration of firewalls and NAT-routers in order to work properly. This is not something every user is capable of doing, and it takes a lot of time.

An adequate Mobile Home Access solution should be easy to install and configure for a novice user with only minimum-to-average knowledge of computing and networking.

2.1.2. Resource consumption

VPNs are based on strong encryption (IPSEC can use DES [3] or 3DES) to allow a secure tunnel between devices across the public Internet. For resource constrained mobile devices, strong encryption is often not possible due to the amount of required processing power and memory needed to perform the mathematical calculations. In addition, strong cryptography used in VPNs might be unnecessary overhead for the type of content targeted in this paper, e.g., music, photos, non-confidential documents, etc., where confidentiality is not necessarily required.

An adequate Mobile Home Access solution should be able to adapt to the device in question, from resource-constrained devices like cellular phones, to full-scale PCs, in terms of memory and
processing power requirements, primarily by making the strength of the security facilities adaptable.

2.1.3. Time consuming and user non-friendliness

The establishment of the VPN might be time-consuming for the short and sporadic access pattern described in the paper, e.g., get an MP3 file, save a photo, etc. It may also be quite unpractical when the user has to type quite a lot on the tiny keypad of a mobile device to establish the VPN. More seriously, the intermittent wireless links may cause the breakdown of the VPN quite often.

An adequate Mobile Home Access solution should employ simple semantics for connection-setup and teardown, and hide the details required for realising the connection to the home network (e.g., the actual location of the home network represented by an IP-address).

There exist some VPN solutions based on Secure Sockets Layer (SSL), which simplify some of the challenges described above. With SSL, only one port will be used, thus firewall/NAT-configuration is simplified. However, most of these solutions still suffer from being too cumbersome to configure, and no known solution exists for mobile phones.

2.2. Home access methods

Several ways exist to access the home domain and resources residing there from remote locations. The simplest way would be to install a Web-server at one of the computers at home. This is however not a complete solution to the problem, because access to other computers in the home network is not ensured. There will also be challenges with firewall- and NAT-traversal. Other ways of accessing services at home have been proposed which make use of the Session Initiation Protocol (SIP) and OSGi [4]. Many of the existing initiatives are focused on controlling home appliances, whereas the goal of the solution discussed in this paper is to provide ubiquitous access to the file system. The requirements to processing capabilities and network bandwidth differ dramatically for a message-oriented remote control system and a file access system.

2.3. Universal plug-and-play

One solution to dynamically manage the rules of firewalls/NAT-routers is the use of the Universal Plug-and-Play (UPnP) protocol [5]. By using this protocol, if supported by the firewall/NAT-router, an application on the inside of the home network can specify how incoming traffic should be handled, e.g., that TCP traffic to port 80 should be forwarded to the MHA server. However, this protocol opens the door to serious attacks by hackers, and it is often advised against enabling such functionality.

3. XML Web Services

Today, a new computing model for building distributed software systems is emerging: Web Services. They are a collection of standards for developing, deploying and providing flexible, platform independent services that are distributed through the Internet. A typical Web Service consists of a Web Service client and a Web Service server (which is commonly referred to as the Web Service itself). The client can invoke operations on the server, which in turn returns the result. The minimum set of enabling technologies and standards of a Service-Oriented Architecture (SOA) utilizing Web Services are:

Web Services Description Language (WSDL) [6] – These are XML documents describing how to invoke a service, which includes the methods that can be invoked, their parameters, as well as the return data type.

SOAP [7] – SOAP is the XML-based protocol used for communicating the service invocations between client and server. These invocations follow a request/response programming model, similar to HTTP; HTTP is also the most common transport for SOAP.

One of the strengths of XML Web Services is that all information required to build a client application is contained in the WSDL. Thus, as long as a WSDL is published, anyone can develop new clients accessing an existing Web Service. This is one of the primary reasons for using XML Web Services in the Mobile Home Access solution, because it should support as many types of devices as possible. Thus, MHA clients do not have to be built by one vendor, but could instead be built by the different device manufacturers or other third parties. XML Web services are also platform-independent technologies, which allow for the development of clients for any imaginable type of device.

With XML Web Services, firewall traversals are in most cases implicitly simplified; transport is by default over HTTP on port 80, and HTTP traffic is usually allowed through most firewalls, at least
outward. Using another protocol (or port) will in many cases result in traffic to be blocked by firewalls, in either direction.

4. The home domain

A home network typically exhibits several of the following characteristics:

Firewall between home network and Internet – Most private networks connected to the Internet employ a firewall to increase security and avoid unauthorized access to the network from the Internet. In addition, most networks use a combined solution where the firewall is integrated with a Network Address Translator (NAT) device, which allows the use of private addresses on the computers on the home network.

Several personal computers – It is becoming common to have more than one computer connected to private networks, i.e., each family member is likely to have their own computer connected to the network. Each user has their own personal documents and services which reside on these computers.

Shared computers – In addition to the above, some computers might be shared by several users, where each user has their own private account with their own personal documents and services.

MS Windows or Linux operating systems – Most desktop computers employ either a variant of MS Windows or a Linux distribution.

The above characteristics describe the environment that the Mobile Home Access system must be able to accommodate.

4.1. Firewall and NAT issues

Before accessing the home domain from the outside (i.e., from a mobile phone), a firewall must in most cases be bypassed. Basically, the problem is that packets from the outside in most cases are not accepted, unless:

(a) The packet belongs to an already existing session set up from the inside of the firewall.
(b) Specific rules are in place in the firewall, and the packet matches one such rule. These rules must be explicitly added by the user or the administrator of the firewall.

A NAT router creates a local network domain by allowing translation from locally defined (private) IP addresses to a global IP address. Traffic to and from the Internet uses this global IP address as sender and recipient address, respectively. The NAT device keeps track of the connections from each computer (i.e., IP address) on the LAN, thereby correctly forwarding incoming traffic to the appropriate computer. A NAT device can behave in different ways; generally four types are distinguished [8]. The different NAT behaviours have different consequences for the possibility to traverse the router. This has been taken into consideration in the design of the Mobile Home Access.

To properly forward incoming traffic to a computer on the home network, rules must be in place on the NAT router which specifies what to do with incoming packets (e.g., incoming TCP packets on port 80 is to be forwarded to the computer with IP-address 10.0.0.1 at port 8080).

5. The mobile home access

This section describes the design goals, the overall architecture and primary mechanisms of the Mobile Home Access solution.

5.1. Design goals of mobile home access

Taking into account all the characteristics of existing solutions and the home domain, the Mobile Home Access has defined the following goals:

Simplified access to the home domain – This is the overall goal of the system. This involves challenges with penetrating firewalls and routers on the border between the home network and the Internet.

User friendly – The solution should be easy to install, easy to maintain and easy to use. Existing solutions that theoretically could provide users remote access to documents in the home domain are difficult to install and configure.

Multi-user support – The solution must support multiple users for each home domain. A home domain typically has several users, either using the same or different computers on the home network. Each user should have personalized access to his/her own documents and services.

Access from any type of connected device – Since the number of end-user devices is increasing, and many users of these could benefit from having access to already existing documents and services in the home domain, it must be easy to expand the system to support new types of connected devices.

Adaptable protocol – This goes along with the previous goal; the protocol should restrict the
amount of information exposed due to various device capabilities and limitations in these. For example, when accessing documents through a cellular phone, only a minimum of information should be transferred, whereas when using a Personal Digital Assistant (PDA), additional meta-information could be transferred as well.

Adaptable and adequate level of security – Since the system should be accessible also from limited-resource devices with less processing power, security features should be adaptable. For example, it might not be strictly necessary to use encryption of transferred information, as long as the authentication and authorization processes are strong enough.

Minimize interactions on the restricted bearer – Since information in many cases will be transferred across a wireless bearer (e.g., GPRS, UMTS or WLAN), the number of interactions between the device and the home domain should be minimized, since latency is often very high across these bearers.

Otherwise improve existing solutions – The Mobile Home Access system should otherwise improve on existing solutions, keep processing power requirements on the end-user device on an acceptable level (according to device type) and reduce complexity.

5.2. Global system architecture

Fig. 1 displays the global system architecture of the Mobile Home Access system.

The components of the Mobile Home Access system are:

Mobile Home Access client – This client provides the user with the required functionality to browse documents, download documents and upload documents to the Mobile Home Access server. This client is developed and based on an XML Web Services stub, generated from the description of the Mobile Home Access server. This contributes to making the solution easily extendable for new types of devices.

Mobile Home Access server – The Mobile Home Access server acts as an XML Web Service server externally, and as a CIFS client internally. It receives requests through the Web Services interface and translates these requests to appropriate CIFS requests, redirecting these to the native networked file system of the home network (see Fig. 2). Responses from the networked file system are in turn mapped back to a SOAP message and sent as a response on the Web Services interface back to the Mobile Home Access client.

The description of the two components above are general, but two specific approaches for performing the mapping between CIFS and SOAP have been found suitable for the Mobile Home Access system; reduced mapping mode and tunneling mode. These two approaches are described next.

5.3. Protocol operation – reduced mapping

The design of the reduced mapping mode has been use-case driven. That means, the protocol consists of those messages necessary to fulfill specific user tasks such as browsing directories, downloading and uploading files.

The client thus supports only a minimum of messages, whereas the server is able to interpret these

Fig. 1. The global system architecture of Mobile Home Access.

Fig. 2. Service request and response in Mobile Home Access.
messages as well as the complete CIFS protocol on the other end. An example message flow for requesting a directory listing can be seen in Fig. 3.

The reduced mapping mode is implemented at our laboratory [9], whereas other parts of it are currently under study. An article with an initial study of the solution has been published in [10].

5.4. Protocol operation – tunnelling

The basic functionality of the tunneling mode is to allow existing CIFS clients access to the home domain by tunneling complete CIFS messages through the XML Web Services interface. Only two methods must thus be supported by the XML Web Service; Request (CIFSAttachment) and Response (CIFSAttachment). The request-command will result in a SOAP message (response) with an attachment which is a binary (or potentially base64-encoded) CIFS packet.

The tunneling mode further adds to the flexibility of the system, since it allows devices with advanced resources, like laptops, to have complete access to the home domain as if it were local.

6. Middlebox traversal for MHA

There are some challenges to be met when delivering client–server based services, and in particular peer-to-peer based services. In particular, most servers reside behind firewalls and NAT routers. The common solution is to perform manual configuration of these “middleboxes”, allowing external access to the internal resources from specific IP addresses.

However, this requires much effort and highly competent users. Additionally, if the IP address of the client changes, which indeed is very often the situation for mobile users, a new configuration entry must be added to the middlebox everytime the IP address changes.
A solution for allowing external access to internal resources on a local area network is described in [8]. This mechanism makes use of the characteristics of the TCP and UDP protocols and state-machines, in order to allow a remote client to access a network. The details of the procedure are left out due to space limitations. However, these mechanisms have been adopted and integrated into a standard solution for XML Web Services. The solution is illustrated in Fig. 4.

The component called TravLibServerProxy in the home network keeps a connection towards the TravLibServer component, which is globally accessible. This is done by polling with regular intervals. When a client wants to connect to the home network, it contacts the TravLibServer component, which again notifies the TravLibServerProxy component in the home network. The TravLibServerProxy in the home network then tries to connect to the client, and at the same time the client tries to connect to the home network. Due to the mechanisms described in [8], although depending on the NAT configuration used, a connection will often succeed.

As part of this project, a library for J2ME devices has been developed, which allows easy integration of existing J2ME JSR-172 compliant clients with this system. The only required additional code in the J2ME source code can be seen in Fig. 5. The referred stub is the original JSR-172 stub. The proxy will change the endpoint of this stub to point to localhost at the port where the proxy is listening. When a method is invoked on the JSR-172 client stub, the SOAP message will be intercepted by the proxy, the proxy will initiate the NAT traversal procedure, and upon successful connection establishment towards the MHA server, the SOAP message will be forwarded. Incoming responses will be intercepted by the proxy and forwarded back to the JSR-172 stub.

The TravLibServerProxy also ensures that it is simple to integrate with the MHA Web Service server, no major changes have to be performed.

To make the solution work in close to 100% of time, the TravLibServer should also allow proxying of user data. This will then be the fallback solution when a direct connection between the TravLibClientProxy and the TravLibServerProxy is not possible to establish. This functionality is part of future work.

7. Performance evaluation

Mobile Home Access is a system which typically will be used from devices connected to a wireless network, e.g., General Packet Radio Service (GPRS), Universal Mobile Telecommunication System (UMTS) or Wireless Local Area Networks (WLAN). All these networks have limited bandwidth capacity, and the transfer rate varies from around 40 kbit/s (GPRS), 384 kbit/s (UMTS) to 54 Mbit/s (WLAN). Thus, it is important to consider the amount of data transferred and only transfer what is strictly necessary. In reduced mapping mode of MHA, only the minimum required operations are supported and only a limited amount of the information in CIFS packets are mapped to SOAP messages for transfer over the wireless bearer. Of course, the most ideal situation would be to use a binary, efficient protocol for the access protocol, but since this is impossible while still supporting the other important requirements to the system, it is necessary to study how and if performance can be improved while still using XML Web Services for transport.
In addition to actual achievable bit rate, there is another performance bottleneck which must also be considered. Cellular packet networks (GPRS and UMTS), and wireless networks in general, have relatively high network latency. It is therefore desirable to have as few interactions as possible across the wireless bearer. The Mobile Home Access does this, and as Table 1 shows there is but for the download case a considerable reduction in the amount of interactions across the wireless bearer. This table only illustrates the number of packets in order to highlight the latency involved with each complete operation, and not to benchmark the throughput of the different protocols (thus packet size is considered of no relevance in this particular table).

To reach maximum efficiency in file transfer, it is necessary to analyse the different components of the transfer time. The transfer time can be broken down into the following components:

- **Cellular network latency** – This is the time it takes for a packet to travel from the mobile device, all the way to the MHA server and for the response packet to come back to the mobile device. It is represented by the measured cellular network Round Trip Time (RTT).
- **Local area network latency** – This is the time it takes for a packet to travel from the MHA server, all the way to the server holding a requested resource and for the response packet to come back to the MHA server. It is represented by a measured RTT.
- **Processing time on file server** – This is the time it takes for the file server to read data from storage device.
- **Processing time on Mobile Home Access server** – This is the time it takes for the MHA server to request data from the network, read requested data from the network, compress the data in memory, encode the data in memory and write the data to the network.
- **Processing time on mobile device** – This is the time it takes for the MHA client to request data from the network, read requested data from the network, decode the data in memory, uncompress the data in memory and store the data persistently on the device.

To improve efficiency means minimizing the time spent in any of these components, wherever possible. Cellular network latency and local area network latency are considered unaccessible components (cannot be improved by the MHA design). The processing latency on the file server is controlled by the operating system and hardware resources available, but these are also considered unaccessible. However, the sources of latency on the MHA server and MHA client can partially be accessed and manipulated.

The following are the main hypotheses to be examined in the analysis:

1. Due to relatively high latency involved in each request/response invocation, the throughput rate will be substantially increased with increasing block size for each request/response.
2. Due to processing of content in execution memory (RAM) on the handheld device as well as on the application server, there exists a maximum optimal block size for transfer on each request/response invocation.
3. Compression of content will, in cases where content is not already compressed, improve the net transfer rate, even if it increases the processing time spent on the MHA server and the MHA client.

It is necessary to pose the following additional questions:

- What is the optimal block size?
- How does block size affect the processing time on the client device?
- How does compression/decompression influence on the time spent processing on the limited handheld device as well as on the server?
- Is there a net benefit of using compression, even with the extra processing time required?

All the evaluations are performed using a Nokia 6680 connected to the 3G/UMTS network of Tel- enor Mobil/Norway.

<table>
<thead>
<tr>
<th>Test case</th>
<th>CIFS packets</th>
<th>SOAP packets</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Login</td>
<td>22</td>
<td>8</td>
<td>0.37</td>
</tr>
<tr>
<td>2. Browse</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Browse</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Get file details</td>
<td>122</td>
<td>143 (3000 bytes/request)</td>
<td>1.17</td>
</tr>
<tr>
<td>1. Download file</td>
<td>122</td>
<td>143 (3000 bytes/request)</td>
<td>1.17</td>
</tr>
</tbody>
</table>

Table 1

Comparison of CIFS and SOAP request/responses
7.1. Throughput rates

Effective throughput rates were measured by transfer of a document with compression ratio of approx. 6.5:1. Fig. 6 shows the results, where graphs are shown for transfer with and without compression, for various original binary block sizes (before base64 encoding).

7.2. Round-trip-time (RTT)

The RTT is used to perform the rest of the estimates. Actual round-trip time between the mobile device and the MHA server has been measured and calculated by averaging the time taken to set up TCP-connections with a normal packet sequence (SYN, SYN-ACK, and ACK).

7.3. Processing time on mobile device

Calculation of the processing time spent on the mobile device, in several scenarios with different block sizes and both with and without Gzip compression, was performed as follows.

To avoid interfering with the measurements, estimation was performed on the MHA server side of the connection. The time spent between the last TCP packet in a request (FIN-ACK) and the first packet in the next request (SYN) is believed to be a good enough estimate of the time spent in processing data on the mobile device (see Fig. 7).

Fig. 8 shows the estimate of processing time spent on the mobile device for each request/response. The values are averaged over a large data set and displayed for some block sizes.

The estimate of the processing time on the mobile device shows that the processing time is relatively flat for the case where gzip compression is used, and it is also lower when gzip is used. However, more importantly, for the situation where gzip is not used, the processing time increases for larger block sizes. It does not increase as much when gzip is used. This indicates that the processing time is spent in either the lower network layers or even more likely in the XML Web Services layer (e.g., in the XML parser), at least somewhere in the steps performed prior to decompression.

Initially, it seemed logical that much of the processing time should be spent by the Base64 decoder and in gzip decompression. However, when disabling

---

Fig. 6. Effective throughput rates with and without Gzip compression for various data block sizes.

Fig. 7. Calculation of time spent processing on the mobile device.
the use of the file system on the device (i.e., not writing received data to persistent storage), skipping Base64 decoding and not using gzip, processing time was measured to decrease by 7–15% (compared to Base64 without gzip compression). This suggests that most of the time is spent elsewhere.

7.4. Transfer block size

In theory, the maximum possible block size with the current implementation should be 49275 bytes binary, which yields \(2^{16}\) bytes when base64 encoded \((49275 \times 1.33 = 65536)\) (base64 encoding the data results in an increase of about 33% of the initial size). This is due to the implementation using the Java String class to represent a block of data ready for transfer. However, the limit proved to be even lower, and successful transfers were made up to only approximately 22,000 bytes (before encoded). It seems that the XML parser breaks down when receiving more data than this (this will be subject to further investigation).

This analysis has supported the hypotheses stated, but more interestingly, it shows that adding a compression scheme to the data transfer mechanism has a net benefit (provided the data is not already compressed) and does not yield noticeable processing overhead on the mobile device. Rather, most of the processing time on the mobile device seems to be spent in the network and middleware layers.

The next paragraphs proceed with further examination of this topic, as support for attachments was added to the SOAP implementation. Thus, the XML parser should be relieved from much of its processing, and this should in theory further substantially increase the effective transfer rate.

The following paragraphs discuss results seen from running the Mobile Home Access client in an emulator on a PC in a LAN. Fig. 9 shows the results in a scenario with the following parameters:

- Network speed limited to 112 kbit/s.
- No limitation on the virtual machine (VM) processing speed.
- Transfer block size of 22,000 bytes.

Fig. 10 shows the results in a scenario with the following parameters:

- “Unlimited” network speed (10 Mbit/s LAN).
- No limitation on the VM processing speed.
- Transfer block size of 22,000 bytes.

7.5. SOAP with attachments (SwA)

As Fig. 9 shows, the throughput rate can be increased by approximately 36% by attaching the Base64 encoded data as a MIME multipart according to the SOAP with Attachments specification.
The increase in throughput is due to less overhead in the XML parser alone.

7.6. SOAP with attachments and compression

By using compression of binary data before Base64-encoding, the throughput rate increases by approximately 78% compared to the initial solution. Such an increase is to be expected by uncompressed material, and the tests here are performed by transferring a typical MS Word document.

7.7. SOAP with Binary attachments

Since it is possible to send binary data as a MIME multipart attachment, it is no longer necessary to use
base64-encoding of the data prior to transfer. The estimated improvement by doing this is an increase in throughput rate by approximately 77% compared to the initial solution. This solution is almost identical to the solution using SOAP with attachments and compression. Thus, there is no significant overhead in performing decompression on the client side, at least not for the case where the link is limited to 112 kbit/s.

However, as Fig. 10 shows, when the achievable throughput rate increases dramatically, the solution without compression provides an increase in 116% in the throughput rate compared to solution with Base64-encoded and gzipped attachment. The relative amount of time spent in decompression becomes a concern when the network throughput rate is high. The processing capacity of the device ends up being the bottleneck in the solution, because the network can deliver data faster than the device can decompress it. Skipping compression in such scenarios thus increases the maximum throughput rate.

8. Security considerations

The system described in this paper is primarily meant to be used from wireless devices, although it also includes support for devices with fixed-line connections. Due to information travelling across wireless bearers, and thus possible exposure to eavesdroppers, several aspects of security should be considered in this system. Although, it was stated earlier in this paper that security should be kept to a minimum to avoid excessive resource consumption and complexity of the solution, some security measures are still of high importance. For example, it must be ensured that no other than the proper owner of documents and services gain access to these resources. This is achieved through authentication. This section discusses some security measures that should be considered in future work.

8.1. Points of attack

The MHA system can be attacked at three different places:

1. Between the MHA client and the TravLibServer.
2. Between the MHA server and the TravLibServer.
3. Between the MHA client and the MHA server.

The most crucial point is the connection between the MHA client and the TravLibServer. If the user-name and password (or other credentials) of a user are compromised here, an attacker will be able to connect to the MHA server of this user. Usernames and passwords must thus be sent in plaintext, and a challenge-response mechanism should be used. If the username and password used between the MHA client and the MHA server are the same as the ones used between the MHA client and the TravLibServer, the connection between the MHA client and the MHA server could also compromise a user account in the system. An attacker will not be able to connect directly to the MHA server, because that is only possible after first contacting the TravLibServer. But if the usernames/passwords are the same across these two connections, the attacker can connect to the TravLibServer first, if the location of the appropriate server is known.

The connection between the MHA client and the MHA server is susceptible to eavesdropping of user data. The MHA system is designed to be adaptable with regards to security, so the user could select appropriate encryption of user data when confidential information is transferred across this connection.

8.2. Authentication

Authentication is the first security function required to protect from illegitimate access to personal resources in the MHA system. Several levels of authentication will be taken care of.

1. Authentication of the MHA server towards the TravLibServer – This is to ensure that no one connects their fake MHA server to the system claiming to be a legitimate user. The goal could be to retrieve user credentials or receive data from a user.

2. Authentication of the MHA client towards the TravLibServer – This is to ensure that no one connects their MHA client to the system claiming to be another user, with the goal of gaining access to another user’s resources.

3. Authentication of the MHA client towards the MHA server – This is to ensure that no one connects their MHA client to an MHA server belonging to another user, with the goal of gaining access to another user’s resources.

4. Authentication of the MHA server towards the MHA client – This is to ensure that no one can set up a fake MHA server and make users connect to this server in the belief that this is their personal MHA server. The goal could be to retrieve user credentials.

The authentication procedure is illustrated in Fig. 11.
8.3. Confidentiality

Confidentiality means that sensitive information is not disclosed to unintended parties, i.e., eavesdroppers. With wireless communication, “everybody” can listen on the correct radio frequencies and pick up transferred information. Confidentiality is realised by encrypting the information prior to transmission, and decrypting at the receiving end. However, for GSM/UMTS networks, the data transferred across the wireless bearer is already encrypted. MHA will optionally support additional end-to-end encryption of data based on personalized preference settings by the user, which could be selected according to the type of user data transferred.

9. Conclusion and future work

This paper proposes the use of XML Web Services technology as middleware for mobile services, and in particular describes the case of a remote file access with a system called Mobile Home Access. The system allows access to personal documents and services residing in a personal home domain, from remote locations using either resource-constrained devices like cellular phones, or more powerful devices like laptops, PCs, etc. The system has been designed for simplicity for the user, as well as focusing on the possibility to develop clients for a vast range of different devices, and finally to allow firewall/NAT-traversal without user intervention and complex configuration procedures. The paper also contributes with a performance evaluation of the use of XML Web Services for this system, but which is relevant for mobile applications in general. Security enhancements for the solution are also suggested.

As part of future work, further studies of the security threats to the solution, as well as implementation of measures to avoid those will be performed. Studies of different JSR-172 implementations should also be performed. The failover solution where a globally accessible proxy is used for transferring data between the client and the server will also be further studied, in order to see how this component will affect the overall system performance.

References


Ivar Jørstad received an M.Sc. in Informatics, with specialization in communication systems, from the University of Oslo in December 2002. He received a Ph.D. in Telematics from the Norwegian University of Science and Technology in 2006. The doctoral thesis was focused around the requirements of future service platforms for mobile services with particular focus on personalisation and service continuity. He is currently the CEO of a Norwegian start-up company, Ubisafe AS, which specializes in security solutions for Internet-based services by...
utilizing security mechanisms of the mobile networks. He also supervises students doing their master-studies at the Norwegian University of Science and Technology. His fields of interest include service architectures and platforms for mobile services and applications, mobile distributed computing, Service-Oriented Architectures (SOA), XML Web Services, mobile terminal platforms and middleware, personalisation of mobile services and security of distributed services.

Do Thanh van received a M.Sc. in Electronic and Computer Science from the The Norwegian University of Science and Technology in 1984 and a Ph.D. in Informatics from the University of Oslo in 1997. After seven year at Norsk Data, a minicomputer manufacture in Norway, where he experienced hardware and firmware design for real time systems, he joined Ericsson in Oslo. He participated to several projects on broadband networks. In 2000, he began at the Wireless Communications department of Telenor R&D. He was in charge of Personal Area Networks and Data Applications (PANDA) research activities focusing on one side networking of short-range wireless technologies like Bluetooth, Zigbee, WLAN, etc. and on the other side the development of mobile applications using OSA/Parlay, Session Initiation Protocol (SIP), XML Web services. He was project leader of several EURESCOM projects with the participation of British Telecom, Deustche Telecom, Island Telecom, OTE (Greece) and Czech Telecom.

In 2001, he got a full professorship at the Department for Telematics at the Norwegian University of Science and Technology. He gives lectures on advance topics in mobile communications for graduate students. He also supervises Master and Ph.D. students at his laboratory at Telenor R&D in Oslo. His current activities at Telenor R&D are focused on personalisation of mobile services, identity management and mobile device environments. He is author of over 100 publications at international conferences and journals. He has chaired and participated to the technical committee of numerous international conferences. He has been guest editor to the Telenor’s magazine Telektronikk. He is also inventor of 20 patents and a dozen of pending ones.