Web Server on a SIM Card

Lazaros Kyrillidis, Keith Mayes, and Konstantinos Markantonakis

Abstract — **In this paper we discuss the integration of a web server on a SIM card and we attempt an analysis from a security, management, operation and personalization perspective. A brief representation of the Smart Card Web Server (SCWS) will take place, along with a use case that will help the reader to identify the way that a SCWS can be used in practice, before we reach to a final conclusion.**

Index Terms— **Smart card web server, SIM cards, e-voting, SCWS.**

I. INTRODUCTION

Almost 20 years have passed since the invention and introduction of the World Wide Web. WWW has brought many new possibilities in the way that we communicate, shop, or just have fun. Originally, the web pages offered static content with little (if at all) interaction with the user, but, it was soon realized that there was a need for personalized content and more user interaction. This led to the creation of more complex programs using modern scripting languages, such as Active Server Pages (ASP) and Hypertext Pre-Processor (PHP).

A very important part of the overall system solution is the web server. A web server is a program that is used to create and "serve" web pages. The user creates a request, usually through the use of a web browser, which is captured by the web server, processed and the results delivered back to the user. For the requested content to be created/ delivered, the web server may in turn need to communicate with back-end databases or even with other web servers. The complexity of the process is hidden from the users, so they do not need to be, aware of the infrastructure needed for their requests to be processed.

The 1990's were not only important for the establishment of the WWW. Another very important development was the introduction of GSM mobile telephony. Because earlier systems had suffered from communication interception and fraud, the industry decided that it was time to move towards better standardized and secure solutions. This was supported by advances in smart card technology and in particular led to the introduction of the Subscriber Identity Module (SIM) into GSM mobile phones. SIM cards could provide tamper-resistant storage of cryptographic keys and sensitive information, as well as executing security algorithms. Over time the mobile phones and their SIMs (to a lesser extent) have become more sophisticated offering richer content to the end user.

As both WWW and SIM technologies are now important parts of modern communication, thoughts of integrating the two into one platform started to appear. If the widely used web server could be combined with the massive population of SIM cards, this could lead to interesting new possibilities for the users. However, the case for this is not clearly made as integrating a web server on a SIM card could create potential threats and concerns, not to mention the added management, operation, personalization costs and effort that this would entail.

This paper tries to examine these issues by conducting an analysis of the SCWS, whilst bearing in mind the obstacles set by current technology. We will start by a brief representation of the first attempt of creating a SCWS and we will describe how the SCWS will operate. A use case will then demonstrate how the SCWS could be used in real life and identify the main issues/concerns from a security, management, personalization and operation perspective, before finally a conclusion will be drawn.

II. WEB SERVER ON A SIM CARD

WEBCARD

The first reference of a Web Server on a SIM card (SCWS) is almost a decade ago. It was a paper from the University of Michigan, Center for Information Technology Integration. It related to a research program in collaboration with Schlumberger's Austin Product Center. The product that was delivered was called Webcard [1].

The Webcard had 16KB of EEPROM, 1.2 Kbytes of RAM and implemented the Java Card 2.0 specification. The web server was a java applet. Because of the memory size limitations, the researchers had to implement a working server with the minimum possible functionality. As a result they chose to leave out those parts of the specifications that were not absolutely essential

The applet had the following methods:

- install(), that was called once during the applet initialization process,
- select() that was called every time that and application wanted to use the applet
- process() that dealt with the incoming packets. Every IP packet was encapsulated in an Application Protocol Data Unit (APDU), so that it could be processed by the appropriate on card method.

The developers used a version of UNIX that mounted the smart card into the OS file system name space and a daemon

Manuscript received March 6, 2010.

Lazaros Kyrillidis, Author, is working as Information Security Strategy Consultant for Encode S.A. Mesogeion 182, Cholargos, Athens, Greece (phone: +30-210-6563880; fax: +30-210-6543576; e-mail: lazaroskyr4@yahoo.gr).

Keith Mayes, Author, is the Director of the Smart Card Centre, Royal Holloway, University of London, UK (phone: 01784 414408, e-mail: keith.mayes@rhul.ac.uk).

Konstantinos Markantonakis, Author, is a Reader in the Information Security Group, Royal Holloway, University of London, UK (phone: 01784 414409, e-mail: K.Markantonakis@rhul.ac.uk).

written in C was used to forward the requests to the card. The daemon did not segment packets that were bigger than 256 bytes but simply truncated them. An IP address was assigned to the card and an external router was used to route the packets for that IP to the host that the card was connected to.

OMA Specification

The Open Mobile Alliance (OMA) is a standards body formed to provide standards for the mobile industry to encourage the interoperability of products in order to help decrease operational costs and deliver high quality products [2].

One of the OMA standards is for the web server on a SIM. The first draft for SCWS requirements appeared on 17th November 2004 and the first draft for the SCWS itself on 21st February 2006. The first draft was actually issued on 9th February 2007. On 21st April 2008 the approved version named "OMA-TS_Smartcard_Web_Server_V1_0" was published as the standard for industry to follow in order to produce a SCWS [3] [4] [5].

The most important parts of the SCWS according to the OMA specification are:

- SCWS: A Smart Card Web Server is an HTTP server that is implemented in a smart card. Its main purpose is to allow mobile network operators to offer new and more advanced services to their customers. These services are based on an HTTP communication between an HTTP client on a mobile phone (most probably a browser, but any other implementation of an HTTP client is not forbidden) and a HTTP server in a SIM. This communication must be independent from the protocol that is used between the SIM and the handset.
- SCWS Gateway: This is an entity that is based on the handset. Its main purpose is to provide the connection between the handset based client and the SIM based server. The gateway is mostly needed when the SIM card does not have a TCP/IP stack and the communication takes place with the local transport protocol that most probably will be the Bearer Independent Protocol (BIP) [6]. The gateway would translate the HTTP requests from the HTTP client to local protocol specific commands and vice versa (the responses from the server to HTTP responses to the client). Another important task of the gateway is the enforcement of an access control policy (ACP).
- HTTP Client: This is the entity that is based on the handset and "produces" the HTTP requests that are forwarded to the server either via the gateway (when the BIP or any other local transport protocol is used), or directly through HTTP communication.
- HTTPS Client: This is the same as the previous entity, but with the ability to communicate with the server over Transport Layer Security (TLS) [7].
- SCWS Administration Application: This entity is based on the network provider's facilities. It is used for installation and updating of the SCWS along with providing content to the user through new or updated HTML pages that are based in the SIM card. The communication between the SCWS and the administration application is secure.

SCWS URL, IP, Port numbers

When a web server is placed on a SIM it must be accessible from a browser. In order for a browser to connect to any site we must provide it with a Uniform Resource Locator (URL) [8]. The URL is provided either directly through the address bar or by other means, such as a hyperlink. The same applies for the SCWS. Because we use HTTP access from a browser the format of the URL must not change. OMA defines the following format for the URL:

http://host[:port][abs_path[?<name>=<value>]]
https://host[:port][abs_path[?<name>=<value>]]

The first URL is used for simple HTTP communication, while the second one provides security over TLS. The ports that were first proposed for HTTP and HTTPS were 20080 and 20443 respectively (in comparison with 80 for HTTP and 443 for HTTPS in conventional web servers), but the specification proposes different port numbers. There are two different cases depending on the transport protocol used for the communication between the client and server.

Using the BIP protocol, the hosting device may implement its own HTTP services, so the need for two new port numbers appeared. The port numbers that the Internet Assigned Numbers Authority (IANA) assigned were 3516 ("smartcard Port") for HTTP and 4116 ("smartcard-TLS") for HTTPS. The connection would be established with the loopback IP address (127.0.0.1):

http://127.0.0.1:3516/file1/file2/test.html
https://127.0.0.1:4116/sec_file1/sec_test.html

Using TCP/IP, in the case of a SIM card implementing its own TCP/IP stack and having direct HTTP/HTTPs access (smart cards of this kind will be available from Java Card 3.0 and later), there is no need for new port numbers. So we have 80 for HTTP and 443 for HTTPS. This means that the SIM card will have its own IP address and the loopback address won't be needed.

http://<smart_card_IP>[:80]/file1/file2/test.html
https://smart_card_IP>[:443]/sec_file1/sec_test.html

Communication Protocols

For a browser to communicate with the SCWS there must be a common protocol that both sides can "speak" and understand. If this protocol doesn't exist or the two parties don't speak a common protocol, a "translator" must do this job. For the purpose of the SCWS the translation is done by the gateway. As was mentioned earlier there are two ways for the client to communicate with the server. The first is direct HTTP access or if the smart card doesn't support this, the communication takes place over the local transport protocol between the device and the smart card most, probably BIP. Let's see how these protocols can be used in practice.

BIP Protocol

The BIP protocol allows the smart card to communicate with external entities such as the mobile phone. For the purpose of the SCWS, BIP will be used along with TCP/IP as will be explained later. When a smart card is using the BIP protocol it can have the following modes:

- Client mode: This mode applies when the smart card "wants" to communicate with the secure administration server that is based within the facilities of the network

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

operator. A BIP channel is opened that enables the smart card to act as a client in this communication. Because the smart card cannot "speak" TCP/IP directly with the remote server, a gateway is used that will make the translation between the two communicating entities. The gateway receives requests in BIP format from the smart card and translates them into TCP/IP format. The reverse process takes places when the response from the remote server reaches the phone (from TCP/IP into BIP).

- Server mode: This mode enables the smart card to behave as a server. The communication is between internal entities of the mobile phone (the browser in the handset and the server in the SIM card) and no external entity need interfere. A gateway again exists, that waits for requests to the specified ports of the SCWS (3516 and 4116), translates these requests to BIP commands and passes them to the smart card for processing. The BIP responses are forwarded as HTTP responses to the mobile's browser.

TCP/IP Protocol

In the case that the smart card is able to communicate using the TCP/IP protocol there is no need for a gateway. The browser can directly make requests to the SCWS and the server can respond without translation. This also applies to communication with the remote server. In both cases the smart card will have an IP other than the loopback (or maybe the loopback if this is chosen as the implementation) and the ports on which the SCWS is listening will be port 80 for HTTP and 443 for HTTPs.

Administration Protocols

The messages between the SCWS and the remote server can be exchanged in two different ways: either with the use of a Lightweight Administration Protocol or with the use of a Full Administration Protocol.

The Lightweight Administration Protocol is suitable when the amount of data that must be exchanged is small. In this case the bearer of the command can be a simple SMS, or multiple SMSs for large commands. When the SCWS receives and parses the command, it must respond to the remote server with the expected response, so that the remote server can determine that the command was executed correctly.

On the other hand, the Full Administration Protocol introduces the idea of a card administration agent. This agent is responsible for the encapsulation and the transportation of the HTTP messages, for reconnection when the connection fails and for communication establishment. It behaves a little like a gateway, with the note that it resides inside the smart card and not on the phone. The security for the connection between the agent and the remote server is provided by the use of TLS.

Briefly describing the way that the Full Administration Protocol is used we can state that there must be an open connection between the two entities over a secure channel (PSK-TLS is used for that purpose). The agent makes the first request to the server and the server responds to that request with an administration command encapsulated within an HTTP response. The agent receives this and passes it to the SCWS that parses it as a secure command, because it was previously accepted from the agent. When the command is executed, the SCWS informs the agent, which sends another request to the remote server. If the server has more commands to send, it does so, otherwise it sends a response asking the agent to terminate the connection.

III. USING THE SCWS FOR E-VOTING

To consider the possible usefulness of SCWS, let us consider the following use case example. All the citizens of country X were provided with ID cards that also store two certificates issued by the government that will help the citizens with their business, government or private communications (for a relevant example see [9]). The first of the two certificates is used for authentication and the second one for digital signatures and both of them are associated with their private keys and two PINs that are used for their protection. On the SIM card is also stored the government's public keys so that it is available anytime the user needs it during her electronic communication with the government.

Let us now suppose that the government has arranged with the mobile network providers, that the two certificates with the private keys, the PINs, the citizen's name and ID card number and the government's public keys, are stored in the mobile phone of every citizen.

The day of elections has come in country X and people are voting. The country is advanced in terms of Internet acceptance and usage between the populations, so there is the ability for the users to vote using their mobile phones. The person that wishes to vote, connects to a specific web site and the site provides a link that when clicked "transfers" the user to the SCWS environment. There, the user is prompted to provide the necessary PIN and if this is correct, the user can click on a link that will send the necessary authentication data to the government's website. If the user is authenticated, then voting is permitted. Let us consider the process flow in more detail.

Process Flow

Let $Cert_{A1}$ be the user's public key certificate that will be used for encryption during e-voting process, $Cert_{A2}$ be the user's public key certificate that will be used for digital signatures, P_{A1} the private key and PU_{A1} the public key of the user that is used for the encryption/decryption process and P_{A2} the private key and PU_{A2} the public key of the user that are used for digital signatures. Additionally the government uses $Cert_{B1}$ as its certificate and P_{B1} and PU_{B1} as its private and public key used for the encryption/decryption process and $Cert_{B2}$ as its certificate and P_{B2} and PU_{B2} as its private and public key used for digital signatures. Also the overall process uses a hash function H both on the user's and the government's side. The e-voting procedure is as follows:

- The user opens the handset's browser, enters the URL of the SCWS and clicks on the link "E-Voting".
- The user's name and ID card number are encrypted with PU_{B1}, also hashed (which produces the hash H_A) and then signed with P_{A2} and along with $Cert_{A2}$ are sent to the remote server that hosts the e-voting site.

$(ID_A, NameA)_{PUB1} (H (ID_A, NameA))_{PA2} Cert_{A2}$

$\longrightarrow$ *Voting Server*

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

- The remote server decrypts $(ID_A, NameA)_{PUB1}$ using P_{B1}, extracts PU_{A2} from $Cert_{A2}$, verifies $(H(ID_A,NameA))_{PA2}$ using the extracted PU_{A2} (and gets H_A), hashes the (ID_A, NameA) using H (and gets H_B), and checks H_A against H_B. If the two hashes much each other, the server authenticates the user and may proceed with the rest of the process. Then, it checks if the citizen has voted again and if not it creates a temporary entry in a database to show that the user's voting is in progress.
- After the SCWS/user is authenticated to the server, he is presented with a link that points to the IP of the SCWS. The user clicks on the link and she is transferred to the SCWS environment. At the same time the remote server hashes the (ID_A, NameA), sends it signed with P_{B2} and also sends an encrypted link L which has embedded authentication data that will be used later on from the user (to authenticate herself on the remote site instead of providing a username/password):

$$(H((ID_A, NameA))_{PB2}(L)_{PUA1} \longleftarrow SCWS$$

- The SCWS receives the $(H(ID_A, NameA))_{PB2}$ and verifies it using PU_{B2}. Then it hashes the user's ID and name that are stored on the SIM card with H and if the two hashes much each other, the server is authenticated and the SCWS can now prompt the user to provide the PIN. Additionally, the SCWS decrypts L using P_{A1}.
- The user provides the PIN, it is checked by SCWS and if it is correct the SCWS displays the link L that points to the remote server.
- The user clicks on the link and is authenticated (as a user) to the voting site.
- The user can now browse the voting site and vote. When her voting is done, the permanent entry in the database is updated, to show that the user has voted.

Security/Management Issues Specific to this Use Case

One obvious issue that has to be addressed is why there is a need for SCWS in e-voting or indeed e-shopping? The answer can be based on three reasons.

The first is the simplicity and familiarity of the browser environment. As the Internet is involved more and more within our daily lives, a large part of the population is familiar with browsers on PCs, and so browsing via mobile phones is a natural extension. In fact there is no need for a new phone client program to be created, since there are a number of browsers already available and in use.

The second reason is the security that is needed for applications such as e-voting or e-shopping. Despite the fact that a number of incidents like viruses and worms started affecting the mobile phones during the last few years, they are still considered to be more secure than PCs. However the crucial advantage is that the SCWS will be stored on the most secure environment available in mass production i.e. the SIM smart card. The security that a SIM card may offer can be the perfect environment to store sensitive data such as digital certificates, private and public key pairs and personal data. For e-shopping or e-voting where the personal data stored may be credit card numbers, ID numbers, etc. the tamper resistance of a SIM would offer extremely valuable protection. Even if a phone or SIM card is lost or stolen, it would be very difficult for someone to access or modify the data that is stored on the SIM.

The third reason for using the SCWS is the transparency of the process and simplicity for the end user. In the e-voting example the user is required to click on a couple of links and to remember only a PIN number and not long and sometimes, easily forgettable, usernames and passwords. All the complex authentication and encryption between the SCWS and the remote server takes place without the user's active involvement and so strong algorithms and long keys maybe used. Overall security is improved yet the user does not have to remember anything other than the PIN number and so incidents from having a very strong (but easily forgettable password) written down in plain view are no longer an issue.

A government's attempt to use anything other than the traditional ballot for elections can be difficult for the electorate to accept. A mobile phone may not be accepted as an appropriate medium for such a serious procedure as an election, being associated mostly with talking, sending SMS, playing games or just browsing the Internet. Therefore the government would need to ensure that voters were not only confident in the use of the solution, but also in its security.

An important and not easily resolved issue is how the certificates will be securely installed on the users' phones and who will be trusted to manage the security sensitive data/keys. One might consider for an election that this is the government's responsibility; however mobile network operators fulfill the trusted roll on a day-to-day basis. Technology and customer choice also create problems as users change their phones and SIM cards with regularity whereas elections are usually infrequent.

It should be clear that the overall architecture must provide the most advanced and tamper/attack resistant security, because if there is even the smallest assumption that some part of the solution has been compromised, this may affect the validity of the elections. The message exchange between the voting site and the SCWS and between the SCWS and the user's browser must be extremely secure and the authentication process must be fully in-line with cryptographic best-practice. Using high levels of security may reduce speed, however this is not a major concern, as traditional voting (and queuing) processes tend to be slow.

A critical part of the voting process is authentication and the subsequent control that the user is not able to vote more than once. The use of public key cryptography ([10]) is recommended and already used to some extent in modern phones and SIMs. Public key cryptography can provide data integrity and non-repudiation with the use of digital signatures. If public key cryptography is not used, then each user must be provided with a symmetric secret key (different than all the others stored on the SIM card) that will be pre-shared with the remote government server.

IV. GENERAL SECURITY ISSUES CONCERNING THE SCWS

The Internet is not a secure environment and many malicious people and programs exist to try and exploit unwary users. Trojan horses, rootkits, spyware and viruses of all kinds are being created in order to steal valuable data, corrupt communications,/operation and even destroy information. The SIM card is designed as an attack/tamper-resistant secure platform and so extending its

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

ability to serve HTTP requests for critical applications is a positive step in the fight against malicious attacks.

Secure Communication Channel

The SCWS can communicate in two different ways outside of the SIM card environment; either with the remote server in order to receive updates or other administrative commands, or with the phone's browser to serve the user's requests.

For the SCWS to remote server communication, a secure channel is needed. The security can be provided by the exchange of a symmetric key that is pre-shared between the two entities, which is used for the encryption of the communication [11]. This key must be strong enough so that even if an impostor eavesdrops the communication, it will be very difficult for him to decrypt the exchanged messages. This process also offers mutual authentication, because both entities can reasonably determine that the message came from the other entity, as encrypting the message demonstrates knowledge of the shared secret key.

For the SCWS-browser communication there may be two ways of communicating. The first one is by using the HTTP protocol and the second one by using the HTTPs [7] [12]. If an application on the phone asks for access on a specific resource that needs little or no security, then the communication can take place over HTTP. In the case of an application that wishes to access data that must be protected, at least three defensive measures should be in place. The first measure is provided by passing the data via a secure channel that will provide confidentiality, integrity and authentication (most probably using HTTPs). The second measure is the requirement for a PIN, in order to authenticate the user. The third measure is provided by restricting SCWS access to only specific applications that are trusted to meet pre-determined criteria for secure execution. The last defensive measure can be applied using an ACP.

Data Confidentiality/Integrity

The data that is handled via the SCWS fits into two categories. Data that is stored on the SIM and data that is in transit. For the data stored on the SIM, one can rely on the physical security that the card platform can provide. Extracting data from a modern well-designed smart card is extremely difficult. If someone wants to physically gain access to the data he must use advanced equipment, techniques and expertise in order to avoid the implemented countermeasures. Such attacks may destroy many smart cards and can be very costly and time consuming with no certainty of success. Modern smart cards/SIM cards have in place many countermeasures and have good attack resistance [13].

The data in transit is more exposed to attacks than data stored in the secure environment of the SIM card. Unless protected by the protocol it can be altered, interrupted or simply eavesdropped. Therefore necessary measures must be taken in order to a) avoid these actions and/or b) to detect if an attack takes place.

Alteration of data that is transferred between the SCWS and an outside entity can happen in two ways. The first is to simply alter the data without trying to add some meaningful content, e.g. malicious commands. This action is a kind of interrupted communication where the resulting message has no meaning, with the intention being to "break" the communication. The second attack is more difficult and is about altering the data in transit in order to insert (or sometime delete) some meaningful commands to allow further action against the server. This attack can be avoided (even if the messages are sent as plain text) by adding a MAC at the end of the message. The MAC value can be computed using the pre-shared symmetric key and upon receiving the message, the SIM card can re-compute the MAC and compare it with the received value. The MAC usage applies only when there is a pre-shared key that was securely exchanged. For the communication between the browser or any other trusted application and the SCWS (and if there is no prior key exchange), the use of public key cryptography is an appropriate solution. Instead of a MAC, a digital signature produced by the private key of the sender (either the browser or the SCWS) can be added to the message. This signature can be verified using the public key of the sender. Using public key techniques is advantageous when the prior distribution of symmetric keys is difficult (or impossible).

The confidentiality of data can be ensured by a) encryption with a symmetric key (pre-stored or session key) and/or b) by encrypting the data using the public key of the receiver. In b) the only entity that can decrypt the message is the owner of the private key i.e. the intended recipient. OMA proposes the use of PSK-TLS for confidentiality/integrity between the SCWS and the remote server and public key (and optionally PSK-TLS) for the communication between the SCWS and the various applications on the handset.

Authentication

For authentication purposes, OMA defines as mandatory the use of Basic Authentication and optionally the use of Digest Authentication [14]. However this approach may not provide an acceptable level of security, because of limitations with Basic Authentication. A better approach would be to pass all communication over an encrypted channel (HTTPs).

V. MANAGEMENT ISSUES

It is a very great challenge to manage a program as complicated as a web server. All the different possibilities for setting up and tuning it in order to work as expected provide a "headache" to the administrator. In recent years some programs made their appearance that could help administrators to install a web server with the most common setup as the default (WAMP for Windows, LAMP for Linux). This helped inexperienced administrators to setup not only the web server itself, but also the scripting language (such as PHP or Perl or Python) and the database (MySQL). The approach helps the administrators avoid mistakes such as having "register_globals=on" in PHP (which has enabled attacks in the past) because these administrative programs come with the most up-to-date solutions to known security problems [15]. If an administrator wants to setup the web server completely on his own, he must be aware of all the latest vulnerabilities related with web server setup. Unfortunately, many administrators either forget or simply do not bother to protect their servers, especially when correcting vulnerability may result in a corrupted program.

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

An example is the "register_globals" problem that was mentioned earlier and is found mostly on systems that still use a version of PHP <= 4.3.10. Correcting this issue may cause important core packets like Pear.php to stop working [16].

On the other hand the simplistic SCWS may not suffer from the problems affecting complex web servers, although its set-up must be carefully managed. First of all, installing the SCWS will require extra effort from the network provider. Adding an important program that may open up "paths" for attacks will demand to be treated with very great care. The SCWS will most probably have a common setup for all the user installations and be administered from a central point under the network provider control. Initially, only new SIM cards will have the SCWS pre-installed and this brings an issue about what will happen to the older SIM cards. Whilst some legacy SIMs are manageable in the field it is perhaps unlikely that SCWS capability would be added in this way. It would therefore take some years for the functionality to reach the majority of the user base via natural card replacement, unless there was a compelling economic/business reason to accelerate the roll out.

The installation of the SCWS (and subsequent software patches) on the SIM would be carried out by the network provider who is also responsible for any problems that will affect the functionality of the SIM and mobile phone.

VI. PERSONALIZATION

An important parameter for a successful web service is how easy it is to provide personalized content to the users according to their preferences and interests.

With the coming of Web 2.0, it became evident that the web user will be more involved with the content of the web itself [17]. Social networking sites, blogs and sites that allow the user to interact in great depth, changed the nature of the web and made users more demanding about what a web site should offer them. Web content providers have adapted to this and for example the "per user" content is a common approach for any site that wants to keep its viewers/visitors. This is done mostly through the use of cookies or with other means in order to check that users see their preferred content (and advertisements of course). It is also common practice for the website developers to adapt the sites' content according to the location of the user. An example is the BBC website that appears differently when accessed from the UK or abroad.

The above practices can (and most probably will) also apply to the SCWS. Per user content can be provided by the user's preferences or with cookies or any other means (the use of cookies is unspecified in [5]). One interesting idea will be to "mimic" the idea of per region access and provide information to the user about the current location. Let us suppose that the network provider offers the user a page in the SCWS that contains useful local information such as phone numbers (police, hospitals etc.), tourist information (points of interest, closest hospital, restaurants) or even automatic conversion to the local currency. So if someone visits a foreign country, he can easily have access to information that might otherwise require significant effort to gather. One problem is who will be the creator and/or provider of this content? Would it be the network provider of the home country, the visited network or some other party?

This is not really a technical problem as a java SIM card with Global Platform can already allow third parties to install and run applications on them. The real issues are of a business nature and ensuring that the applications and their owners/managers do not undermine the security of the SIM platform.

VII. WEB SERVER ADMINISTRATION

Administering a web server is a difficult task that requires responsibility and technical knowledge. The administrator must know how to setup the server, how to secure it, what measures must be taken in order for the server to work 24/7 and how to setup virtual hosts that will allow multiple sites to be served by a single IP address. Most probably he is required to install one or more server-side languages in order for dynamic sites to be served and must take care of the file system where the web pages are stored (access rights etc.). An administrator may also be responsible for many other things that require a web server to run smoothly (patches, updates, new software versions, DNS) and all these responsibilities justify the need for technical expertise in order for a web server to operate without problems.

In the case of the SCWS, management is the mobile network provider's responsibility. There may be a simple administration page that will be used to download the new versions of the installed applications, but the network provider will be able to manage the core functionality of the SCWS remotely.

VIII. COMMUNICATION CHANNEL SPEED

The speed of a web server is of huge importance. The users are not concerned if one or one thousand machines in parallel process their requests; instead they care about the overall response time. Traditional web servers can be extremely fast, but this speed depends not only on the server hardware capability, but also on the speed of the communications channel that carries the requests/responses. If there is an extremely fast cluster of computers serving web requests, but the capacity of the connecting line to the network is very small, then there is no way that the processing power of this cluster can be exploited. From the user's perspective the overall performance will be poor.

In the SCWS context the speed provided between the handset and the SIM card, may not be ideal [18]. The traditional ISO 7816 interface that connects the SIM card with the phone is too slow to serve the incoming requests and outgoing responses. Therefore the need for a high-speed interface was recognized by ETSI in the standard for the SIM USB interface (8Mb/s) that will replace the older ISO 7816 interface [19]. This means that the necessary speed for the SCWS communication channel will be provided only when the USB interface is widely available. An interesting point to note is that traditionally you could have millions of users accessing one web site whereas with SCWS you have one website per user.

IX. IP MOBILITY

Another very important issue is how the server will be accessed. A traditional web server most probably has a static IP so that it can easily be located by DNS servers [20]. For

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

example, a URL www.xyz.com will most probably be mapped to an IP like 194.12.34.12, so when a user enters the above URL in the browser's address bar, the mapping is done easily and quickly.

For the SCWS the above procedure becomes more difficult[1]. Handsets are mobile devices; they are transferred between cities, countries and continents. If the SCWS is used for serving requests initiated only from the phone's browser, then the IP will not have to change and will be accessible only to the browser. However, if the SCWS becomes accessible to outside entities, how will the problem of the IP mobility be resolved? IP address ranges are applied to cities or countries, so if someone is connected to the Internet from a specific city, his computer will have an IP address between the IP address range for that city and this IP will change if he changes the place that he uses for connection. While this does not affect users that simply browse the web, it will be difficult for a SCWS to serve requests from users that want to access it. In order for this issue to be addressed, there must be a mechanism that maps a specific IP address to a phone and is never changed, regardless of the phone's location. The answer to this problem can be found within RFC3344 and RFC3775, for mobile IPv4 and IPv6 respectively. Briefly, these two RFCs define two addresses for the mobile node; its home address and its care-of address. The packets that are destined for the node's home address are routed to its current care-of address (that is based on the node's current location) and this binding enables the node to move without any problems [21][22].

X. CONCLUSION

The future of the SCWS is at least interesting. Java Card 3.0 should overcome some obstacles that are associated with current card platform technology. With the TCP/IP stack that will be implemented in the SIM card and the direct HTTP access that this will offer, the SIM card will have an IP address that can be accessed from outside of the phone. The trend towards increased memory capacity will enable richer applications to be implemented and the USB connection will make the delivering of the content much faster.

To make a prediction, the future of the SCWS will be determined by at least two things: a) the network providers' willingness to use it, which will be decided by the market/profit perspectives and the user acceptance, and b) the technology obstacles that are currently in place. The latter point should resolve itself as technology is moving ever forward, and eventually it should be possible for the SCWS to have all the necessary processing power and memory capacity needed with fast communication via HTTPs over a USB interface.

REFERENCES

[1] Jim Rees, Peter Honeyman, *Webcard: a Java Card Web Server*, Center for Information Technology Integration, University of Michigan, 1999
[2] http://www.openmobilealliance.org/
[3] OMA, Enabler Release Definition for Smartcard-Web-Server Approved Version 1.0-21 April 2008, OMA-ERELD-Smartcard_Web_Server_V1_0-20080421-A
[4] OMA, Smartcard Web Server Enabler Architecture Server Approved Version 1.0-21 April 2008, OMA-AD-Smartcard_Web_Server_V1_0-20080421-A
[5] OMA, Smartcard-Web-Server Approved Version 1.0-21 April 2008 OMA-TS-Smartcard_Web_Server_V1_0-20080421-A
[6] ETSI TS 102 223
[7] Internet Engineering Task Force, *HTTP over TLS* http://www.ietf.org/rfc/rfc2818.txt
[8] Internet Engineering Task Force, *Uniform Resource Locators (URL)* http://www.ietf.org/rfc/rfc1738.txt
[9] AS Sertifitseerimiskeskus, *The Estonian ID Card and Digital Signature Concept Principles and Solutions*, Version 20030307
[10] W. Diffie and M. Hellman. *Multiuser cryptographic techniques.* In Proceedings of AFIPS 1976 NCC, pages 109–112. AFIPS Press, Montvale, N.J., 1976
[11] Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone "*Handbook of Applied Cryptography*", August, 1996, pp. 15–23, 352-359.
[12] Internet Engineering Task Force, *Hypertext Transfer Protocol (HTTP 1.1)*, http://www.ietf.org/rfc/rfc2616.txt
[13] Wolfgang Rankl and Wolfgang Effing "*Smart Card Handbook, Third Edition*", 2003, John Wiley & Sons, Ltd, pp. 521–563.
[14] Internet Engineering Task Force, *HTTP Authentication: Basic and Digest Access Authentication*, http://www.ietf.org/rfc/rfc2617.txt
[15] Chris Shiflett, *Essential PHP Security*, O'Reilly, October 2005
[16] Stefan Esser, *$GLOBALS Overwrite and it's Consequences* http://www.hardened-php.net/globals-problem, November 2005
[17] Paul Anderson, *What is Web 2.0? Ideas, Technologies and Implications for Education*, Technology & Standards Watch, February 2007
[18] Keith Mayes, Konstantinos Markantonakis, *Smart Cards, Tokens, Security And Applications,* Springer, 2008, pp.62-63
[19] ETSI SCP Rel.7
[20] Whil Hentzen, *DNS Explained*, Hentzenwerke Publishing, Inc., p.3-5
[21] Internet Engineering Task Force, *IP Mobility Support for IPv4,* http://www.ietf.org/rfc/rfc3344.txt
[22] Internet Engineering Task Force, *Mobility Support in IPv6*, http://www.ietf.org/rfc/rfc3775.txt

1. It must be mentioned that the IP issues must be considered only when the SIM card acquires an IP stack and an IP address and will stop responding to requests that are only targeted to the loopback address (127.0.0.1).

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)