The GAL Monitoring Concept for Distributed AAL Platforms

Felix Büsching, Maximiliano Bottazzi, and Lars Wolf
Technische Universität Braunschweig
Institute of Operating Systems and Computer Networks (IBR)
Email: {buesching | bottazzi | wolf}@ibr.cs.tu-bs.de

***Abstract*—In most of today's healthcare and AAL scenarios a multitude of data is recorded by different sensors for various purposes. By definition, this data is mostly personal and can be assigned to certain persons, which again makes it interesting for shady data miners, hackers, or even health insurance companies. Thus, this data needs special care regarding privacy and security.**

On the other hand, if the systems are spread over a larger area, a remote access to such systems is essential for the operator. At least a basic system monitoring should be possible and backups to a distant location are beneficial if the distributed systems fail. In this paper we provide concepts and solutions how in distributed installations of personal AAL platforms privacy and security can be preserved, while remote backup and restore functionality and remote system monitoring can be provided.

I. Introduction

A huge amount of personal data is generated in Ambient Assistant Living (AAL) and healthcare applications. Many different supporting techniques rely on data gathered from many different sensors for various purposes. E.g., while most fall detection algorithms [1] only base on data from the past few seconds, a fall risk assessment [2] may need data from a longer period of time, and when performing Activity of Daily Living (ADL) monitoring [3] nearly everything has to be stored for further processing and analyses.

In many scientific AAL projects issues of data security and privacy are not widely addressed, which is surely OK for pure research projects, in which often every single bit is stored for good for further analysis, because "if we knew what it was we were doing, it would not be called research, would it?"[1] On the other hand, in products or applications *for the real world*, data avoidance and data minimization are mandatory. When performing field studies, these issues will have to be addressed, because *real* people with non-negligible concerns and healthy suspicion are involved. Additionally, in many countries ethical review committees have to approve these field studies to be safe and harmless for the participants.

In contradiction to that, it is necessary to have some kind of remote system monitoring and control capabilities to reduce the number of disturbing personal visits at the participant's homes in order to just verify the functionality of the system. Thus, remote access to the user's system is also desirable.

That leads to the question: Are privacy and system management competing objectives or can booth be achieved in a concurrent way?

The rest of the paper is structured as follows. After we shortly describe the GAL project, the MSHP, and the deployment plan for the field studies in section II, our privacy and security paradigm is explained in Section III. In Section IV concepts and solutions for remote monitoring and backup/restore strategies are given. The implemented and evaluated system is described in Section V. We conclude the paper in Section VI.

[1] Albert Einstein

II. Motivation: The GAL Project

In the "Lower Saxony research network *Design of Environments for Aging* (GAL)" [4], like in many other AAL related projects, new applications of information and communication technology are to be identified, enhanced and evaluated. This project's focus is on single elderly persons living at home alone.

Subsets of the sensors for following applications have been and will be installed during field tests:

- A **fall detection** by worn accelerometers (based on INGA [5]), by a camera, and by microphones.
- A **fall prevention** through a gait analysis [2] by the above sensors and an additional laser scanner.
- **Monitoring of Activities of Daily Living (ADL)** [3] by an additional integration of home automation sensors (switches, door contacts, motion detectors, etc.), current sensors, "intelligent" white goods, and ultrasonic sensors.

Prototypes have already been integrated and systematically evaluated in concrete living and domestic environments. For the near future field studies with a double-digit of participants are planned in which many people and appropriate apartments are to be equipped with many sensors.

Unlike e.g. in the WASP project [6], where data is processed remotely, in GAL the processing is done on a local platform within the inhabitants home.

A. Multi Services Home Platform

The Multi Services Home Platform (MSHP) is located in a person's flat and belongs to the inhabitant. In general it is a standard PC or a Set-top box [7], running an OSGi based middleware and integrating many features [4]. In Figure 1 a

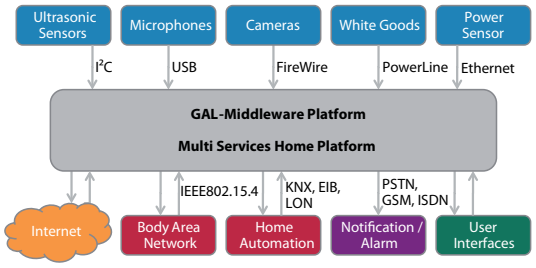

Figure 1: The Multi Services Home Platform (MSHP).

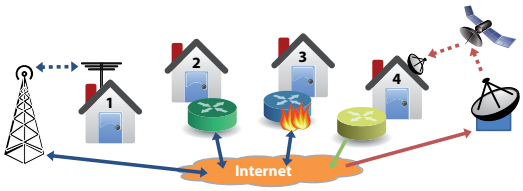

Figure 2: Scattered deployment in different locations with various connections to the Internet.

diagram of a MSHP with exemplary interfaces to various sensors, actuators, and user interfaces is given. On this platform, all data is stored, aggregated, fused and processed. The user can interact via various user interfaces and alarm signals are forwarded through different channels, including phone calls, SMS, and E-Mail via an Internet connection.

B. Deployment Plan

In a field study, 30 apartments in 3 cities will be equipped with MSHP installations with varying sets of sensors. An existent Internet connection cannot be assumed for every single apartment, but it should be utilized when present. For places without an existing Internet connection at least GRPS network coverage (or better) can be assumed, so that in any case 2/3G modems should be able to establish a connection to the Internet.

In Figure 2 some of the supposable scenarios for the various Internet connections are given. "Normal" xDSL, ISDN or POTS lines may be routed (2) – most commonly with Network Address Translation (NAT) – and (additionally) secured by a firewall (3); according to the used technology they differ in terms of throughput and latency, whereas usually asymmetric connections are common (downstream » upstream). In satellite connections (4) normally separated channels for up- and downstream are used, whereas the upstream is usually established via a cabled connection (low bandwidth, low latency) and the downstream is realized via a satellite (high bandwidth, high latency). For pure wireless links (1) normally a 2/3/4G cell phone network is used – with varying speed and quality; some provider only offer a private IP address so that for in-house routing a second NAT is needed.

III. GAL Privacy and Security Paradigm

Dealing with personal data like activity data or vital parameters requires a high degree of considerateness regarding privacy and security. It must be avoided in any case that people feel alienated, because they do not know who is processing their data for what purpose. Thus, the overall privacy paradigm is simple, but convincing:

- All recorded data belong to the person who generated it.
- The data stays on a local platform, the user owns.
- No low-level data, such as single sensor values or vital parameters leave the platform (without permission of the owner).
- All processing of the data is done locally on the user's platform.
- Only processed and aggregated high-level events or alarms like a *detected fall* or an *unhealthy behavior* is reported to the outside (health-care professionals, relatives, etc.).
- From the outside the local system must not be accessible.

From a privacy and security point of view, this sounds reasonable; from an administrative point of view, it may sound horrible at first glance. In a distributed and scattered deployment like described in II-B, it is essential to at least have some basic monitoring capabilities. Otherwise visiting the people is probably the only way to find out whether the system is up and running. Thus, the privacy and security paradigm seems to prohibit remote configuration, monitoring, or backup and restore functionalities.

IV. Platform Monitoring

While backups may be subordinate in short term studies, monitoring is essential for a scattered and distributed deployment. Monitoring a system means not controlling a system. In fact, monitoring comes first and when monitoring detects misbehavior or a failure, the controlling of the system can be the next step to try to fix the failure. For a system administrator it would surely be desirable to have a full (root) *remote* access to all areas of the system, but providing such an access would definitively ruin the privacy and security paradigm. Nevertheless, while still in the stage of a research project and way of alpha or beta stadium it has to be considered, whether such an access can be granted somehow, but we will come to that at the end of the Section. Note that for a skilled person it is only a question of time to gain root access to a system to which he/she has *local* and *physical* access.

A. Simple Heartbeat Messages

The seemingly first and easiest step towards a remote monitoring is a simple heartbeat message. One system just "pings" another system at regular intervals to tell that it is still alive and working. As sending and receiving *ping* (ICMP echo request) messages is part in nearly every operation system, this functionality can easily be implemented. At a central monitoring instance where the heartbeat messages from the distributed MSHP systems are received, an administrator could gain a rough overview of the scattered systems.

A systemic shortcoming of such heartbeats is the low information content. The only information is that the system is powered up, the Internet connection is working, and the process generating heartbeat messages has not crashed yet. On the other hand, such a simple strategy can easily go in hand with the privacy paradigm as long as the MSHP only sends these messages and not listens to replies.

But, there are security issues as well: Standard ICMP messages are transmitted in plain text and thus not only the central monitoring instance is able to receive the heartbeats – any system on the path can read that (admittedly not very worthy) information. The bigger additional risk is that the sending of ICMP messages requires an access (at IP level) to the Internet, which again leads to the demand for a well configured and maintained firewall and a frequently updated system to constantly close possible security holes in every part of the system.

B. Securing the System

It is surely possible to configure a firewall to only allow outgoing ICMP messages, but that is not very flexible either. For a more complex monitoring scenario or for a future remote configuration and control of the system a smarter solution is needed. From the privacy and security paradigm we know that the system must not be accessible from the outside. Thus,

all communication must be initiated by the system itself. This is also beneficial when looking at the different Internet connections of the deployment (Figure 2): A system behind a NAT or a firewall can only be reached from the outside with respectively configured forwarding mechanisms in the router/firewall; this may (with some effort) be possible for most home routers – but impossible when dealing with a second NAT at the provider side by a 3G connection. Thus, the establishment of the connection should be performed by the distributed systems anyway.

While one end of the communication channel is the particular MSHP, the other end should be a centralized facility, where the data of all MSHP to monitor is collected and presented.

Realizing a communication of integrity over insecure and uncontrollable channels like the Internet requires an authentication of the involved partners and an encryption of the data. This can either be achieved by specialized protocols like SSH or HTTPS or by securing all traffic by tunneling a through Virtual Private Network (VPN).

C. VPN Considerations

A VPN can provide a tunnel through an unreliable an insecure network like e.g. the Internet. All VPN solutions have in common, that data is encapsulated at the entry point to the tunnel and expanded at the exit point. This encapsulation always implies an overhead. In [8] the major implementations are compared.

The Layer 2 Transport Protocol (L2TP) [9], Internet Protocol Security (IPSec) [10], and the Point-to-point Tunneling Protocol (PPTP) [11] are well established (and pretty old) Internet standards. The developer of PPTP (Microsoft) nowadays discourages from using it because of major security flaws. As L2TP is just responsible for tunneling and not for encryption, it is often used in combination with IPsec. IPsec itself is located at the network layer (Layer 3) and by that not easy to integrate in existing networks. Especially in changing or mobile networks, L2TP and IPsec are complicated and hard to setup. Peer-to-peer VPNs like tinc[2] do not meet the demands of our centralized monitoring approach.

After reviewing major VPN solutions, we decided to use OpenVPN[3] for our purposes. OpenVPN is a so-called Secure Socket Layer (SSL) VPN, utilizing Transport Layer Security (TLS) [12], which is as well an Internet standard. It utilizes the free OpenSSL[4] implementation. Both is open source software and no license fees have to be paid. In Figure 3 the basic functionality of OpenVPN is given. OpenVPN provides a virtual network adapter (like an additional Ethernet interface) to the operating system. All network traffic using this interface is then encapsulated and forwarded through a secure tunnel. The tunnel is established between an OpenVPN client (MSHP) and a server, the VPN concentrator. The concentrator again also has virtual network interfaces – one for every connected client. While configuring such a VPN it can be decided whether the traffic of the connected client shall be bridged (the clients can exchange data on layer 2 basis) or routed (the routing instance of the concentrator decides). As a communication between the particular MSHPs is not intended, the routed interface should be configured with no routes between the singular MSHPs.

[2]http://www.tinc-vpn.org/
[3]http://www.openvpn.net
[4]http://www.openssl.org/

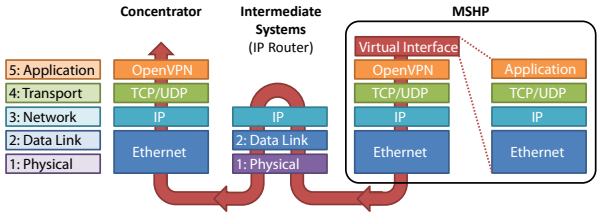

Figure 3: OpenVPN functionality.

For each MSHP installation there are just two easy rules to be observed to keep such a system secure from attacks:

- From the outside, nothing is accessible. The only application that is allowed to use the regular network interface is the OpenVPN process – for outgoing connections only.
- All other processes (whatever is intended: Heartbeat sending, SSH server, etc.) are only bound to the virtual VPN interface and thus only able to communicate with the concentrator.

When configured correctly this solution works with any possible Internet connection as application layer VPN is used. Thus, the MSHP can be preconfigured before installation; after being powered on and connected to any kind of Internet connection, it will establish a secure tunnel to the concentrator which then can start monitoring the system.

D. Professional System Monitoring

Once a safe and secure connection from a MSHP via a VPN to a concentrator is established, this connection can surely be used for more than just simple heartbeat messages – if necessary. A common and widely used protocol for system monitoring is the Simple Network Management Protocol (SNMP) [13], that currently exists in 3 versions and is implemented in most professional network devices. SNMP is a powerful protocol and to enumerate all its possible usages is beyond the scope of this paper. The first two versions (SNMP v1 and v2c) provide no security mechanisms at all, but they are the ones with the widest distribution – therefore an implicit securing by a VPN tunnel is a good and easy way to use these protocols for the monitoring of distant devices.

Note that SNMP (if configured that way) may also allow write access to the supported devices, which may be the first small step towards remote configuration and the first big step to undermine our privacy paradigm.

E. Remote Monitoring and Control – Summary

In Figure 4 the basics of the remote monitoring (and control) system are summarized. The distributed MSHP systems set up a VPN tunnel to a concentrator, whereas the communication channel can be assumed as safe and secure. Only the VPN client is able to access the external interface and every communication that leaves the MSHP system is encapsulated.

The VPN concentrator acts as counterpart for the VPN tunnels. Here, the unencrypted monitoring data is available to any kind of monitoring application. When implemented, control mechanisms are also present here. It has to be ensured that only authorized personal has access to the VPN concentrator. This can be achieved by common access control mechanisms like allowing only *local logins* and having a *physical access control system* for the room or building the concentrator is located.

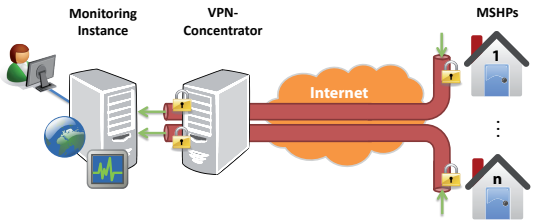

Figure 4: Monitoring over a Virtual Private Network.

For pure monitoring solutions, the monitoring instance could additionally provide the status data to a network by pushing the data e.g. to a secure web server.

V. Implementation and Evaluation

For the intended field studies, we implemented the described system: The central monitoring facility and the VPN concentrator are located in a data center at our university. The VPN concentrator has a fixed and public IP address so that there is no need for the client MSHPs to rely on any kind of Domain Name Service (DNS). We configured the OpenVPN service in *routing* mode with no routes set, so that misbehaving systems cannot effect the other systems. For the authentication of the VPN connection we generated pairs of certificates in advance and installed them on the MSHP systems and the concentrator. For the deployment that brings two advantages: On the one hand, the completely configured system can just be carried to its intended destination at an elderly person's apartment; after being switched on and connected to the present Internet connection no further action is needed for the monitoring. On the other hand, if a MSHP was somehow corrupted or stolen, the respective certificate can easily be deleted from the concentrator and this system will no longer be able to connect. This procedure works well for any investigated Internet connection and the only observed limitations are in the throughput and stability of the Internet connection, which we cannot influence. For monitoring purposes any available Internet connection is sufficient as the generated traffic is low compared to even the lowest available upstream bandwidth. Even remote system administration underlies no shortcomings – as long as no graphical user interface is used. We measured the maximum needed bandwidth for a SSH connection to be 2.8 kbyte/s for the uncommon case of constantly transferring data. However, when a graphical interface for administration like a VNC [14] connection is used, surely a higher bandwidth is needed – we measured a maximum bandwidth of 51.2 kbit/s (average at 13.1 kbit/s) for a VNC forwarding of a 800x600 display.

VI. Summary and Conclusion

After we have stated the project's privacy and security paradigms, which we claim to be universal for many AAL related projects and products, we stated Virtual Private Networks being well suited for a centralized and secure remote monitoring of distributed systems. Every single system sets up a separated authenticated and secured tunnel for communication between the scattered AAL platforms and a VPN concentrator at a central point. By this strategy even insecure management protocols like SNMP can be used to monitor and configure distant systems.

We have determined OpenVPN as an adequate solution, because it is easy to setup, and (as an application layer VPN) very flexible in terms of the underlying Internet connection. In fact, with this solution, nearly every Internet connection can be utilized for a secure remote monitoring and management.

We have evaluated that preconfigured platforms with preinstalled certificates can easily be deployed in a distributed setup. We have also evaluated that for pure monitoring purposes every Internet connection is sufficient as the expected network traffic for monitoring purposes is low in comparison to any available upstream capacity.

Thus, with the presented strategies and concepts remote monitoring and even a remote configuration of distributed AAL systems can be achieved, without significantly undermining elemental demands for privacy and security.

Acknowledgment

The Lower Saxony research network "Design of Environments for Ageing" acknowledges the support of the Lower Saxony Ministry of Science and Culture through the "Niedersächsisches Vorab" grant programme (grant ZN 2701).

References

[1] F. Bianchi, S. Redmond, M. Narayanan, S. Cerutti, B. Celler, and N. Lovell, "Falls event detection using triaxial accelerometry and barometric pressure measurement," in *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, sept. 2009, pp. 6111 –6114.

[2] M. Marschollek, K.-H. Wolf, M. Gietzelt, G. Nemitz, H. Meyer zu Schwabedissen, and R. Haux, "Assessing elderly persons' fall risk using spectral analysis on accelerometric data - a clinical evaluation study," in *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE*, 2008, pp. 3682 –3685.

[3] A. Hein, S. Winkelbach, B. Martens, O. Wilken, M. Eichelberg, J. Spehr, M. Gietzelt, K. H. Wolf, F. Busching, M. Hulsken-Giesler, M. Meis, and P. Okken, "Monitoring systems for the support of home care," *Inform Health Soc Care*, vol. 35, no. 3-4, pp. 157–176, 2010.

[4] M. Eichelberg, A. Hein, F. Büsching, and L. Wolf, "The GAL middleware platform for AAL," in *e-Health Networking Applications and Services (Healthcom), 12th IEEE International Conference on*, 2010.

[5] F. Büsching, U. Kulau, and L. Wolf, "Demo: Inga: an inexpensive node for general applications," in *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '11. Seattle, WA, USA: ACM, 2011, pp. 435–436.

[6] L. Atallah, B. Lo, G.-Z. Yang, and F. Siegemund, "Wirelessly accessible sensor populations (WASP) for elderly care monitoring," in *Pervasive Computing Technologies for Healthcare, 2008. PervasiveHealth 2008. Second International Conference on*, 30 2008-feb. 1 2008, pp. 2 –7.

[7] F. Büsching, M. Doering, and L. Wolf, "Integration of an "environments for ageing"-platform in soho-routers," in *Consumer Electronics (ISCE), 2010 IEEE 14th International Symposium on*, june 2010, pp. 1 –6.

[8] S. Khanvilkar and A. Khokhar, "Virtual private networks: an overview with performance evaluation," *Communications Magazine, IEEE*, vol. 42, no. 10, pp. 146 – 154, oct. 2004.

[9] W. Townsley, A. Valencia, A. Rubens, G. Pall, G. Zorn, and B. Palter, "Layer Two Tunneling Protocol "L2TP"," RFC 2661 (Proposed Standard), Internet Engineering Task Force, Aug. 1999.

[10] B. Patel, B. Aboba, W. Dixon, G. Zorn, and S. Booth, "Securing L2TP using IPsec," RFC 3193 (Proposed Standard), Internet Engineering Task Force, Nov. 2001.

[11] K. Hamzeh, G. Pall, W. Verthein, J. Taarud, W. Little, and G. Zorn, "Point-to-Point Tunneling Protocol (PPTP)," RFC 2637 (Informational), Internet Engineering Task Force, Jul. 1999.

[12] T. Dierks and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2," RFC 5246 (Proposed Standard), Internet Engineering Task Force, Aug. 2008.

[13] D. Levi, P. Meyer, and B. Stewart, "Simple Network Management Protocol (SNMP) Applications," RFC 3413 (Standard), Internet Engineering Task Force, Dec. 2002.

[14] T. Richardson, Q. Stafford-Fraser, K. Wood, and A. Hopper, "Virtual network computing," *Internet Computing, IEEE*, vol. 2, 1998.