A Scalability Study of AAA Support in Heterogeneous Networking Environments with Global Roaming Support

Daniel Granlund, Christer Åhlund
Luleå University of Technology
Division of Pervasive and Mobile Computing
SE-931 87 Skellefteå, Sweden
{daniel.granlund, christer.ahlund}@ltu.se

Abstract—In this paper, we present a scalability study of AAA support in mobile heterogeneous access networks with respect to server and network load related to AAA processes using the RADIUS protocol. Technologies such as IEEE 802.11, CDMA 2000 and UMTS which all support the RADIUS protocol for AAA handling are discussed and analyzed. Typical performance data are gathered and complemented with a theoretical study in order to achieve an overview of what parameters will affect the performance and scalability of the network. Also, guidelines are developed for network design in order to achieve the desired performance for a given number of users. Results of this study include the conclusion that the main bottleneck of the AAA procedure is not necessarily the AAA server CPU power. Aside the cases with a high proportion of computationally intensive WiFi sign-ons with strong encryption, performance issues may be caused by AAA server network connection bandwidth, and RAM memory. In cases where a high number of users reside in the same user database, database performance becomes a significant issue. In order to achieve better performance, CPU load balancing over several servers may be performed.

Keywords: AAA, Scalability, Performance, RADIUS, Mobility

I. INTRODUCTION

The increasing demand for seamless mobility along with reliability and security in heterogeneous networking environments increase the need for mobility management and Authentication, Authorization, and Accounting (AAA) handling mechanisms. Challenges include providing low handover latencies along with low signaling overhead and good scalability. Over the last couple of years, a large number of published results have been presented in this area and several standards exist to support needed functions for most needs.

Mobile IP (MIP) [1] and Session Initiation Protocol (SIP) [2] are well known protocols to support mobility management on the network and application layer respectively. However, security and scalability are often mentioned and treated as a

separate issue or mentioned as future work. In practice, the AAA handling is typically managed by each access technology and service provider individually. In a heterogeneous access scenario this may cause inconveniences on many levels since compatibility and interoperability issues are likely to occur.

The preferred solution would be to provide a uniform way of handling AAA related tasks, regardless of access technology or service type. In previous work [3],[4] we presented such a scheme where a connection between RADIUS [5] and DHCP was created in order to support IP mobility between WiFi and CDMA 2000 networks with a common AAA infrastructure. The most common implementation of RADIUS supported AAA handling in WiFi networks is referred to as WPA2 Enterprise [6]. WPA2 Enterprise implements most of the IEEE 802.11i standard which covers both user authentication and data privacy.

A larger implementation of WPA2 Enterprise can be seen in the EduRoam [7] federation which is a global roaming service, providing the ability for staff and students within the education and research community to use their home institution credentials to gain access to a service at a visited institution. The EduRoam infrastructure uses RADIUS servers which are interconnected in a hierarchical manner in three distinct layers. A Confederation top-level server which is at the root of the tree, then there are Federation top-level servers, typically for each country and finally Institutional level servers placed at each institution.

Interconnecting servers in this type of large, globally spanning network, handling millions of users raises the question of overall performance and scalability of the system. This paper investigates the performance of standard AAA server and provides a study of the scalability of a large scale system.

The rest of the paper is organized as follows: Section 2 provides background and a more detailed specification of the issue being addressed. Section 3 describes parameters affecting

AAA system performance along with theoretical discussion. Section 4 presents experimental setup, results of experiments and calculations and finally, Section 5 concludes the paper.

II. ARCHITECTURE DESCRIPTION AND BACKGROUND

Previous work includes the design of mobility management scheme for heterogeneous access among different wireless as well as wired networks [3]. The proposed architecture supports IPv4 and is implementable in a wide range of networks. IP mobility is supported by tunneling packets to the home network.

Figure 1. System architecture

Figure 1 depicts the overall architecture with CDMA2000 and WiFi as example access networks. The AAA handling is described in [4] where a custom made feature has been implemented in the AAA servers in order to provide IP configuration from the home network by creating a dynamic connection from the AAA server to a local DHCP server. This behavior is already supported by PPP based technologies and the purpose of the specialized implementation was to achieve the same handling for Ethernet based connections.

In a typical case, a mobile user may choose to enter an arbitrary WiFi cell and will, upon association with the access point be prompted for login information. The login username has the form of *user@realm* where the realm part is used to indicate to which home network (or provider) the user belongs. If the home realm is different from the local realm the local AAA (AAA-L) server will proxy the request, possibly by a number of intermediate servers to the home network AAA sever (AAA-H) where the actual authentication takes place. Upon a successful authentication the AAA-H will provide a reply message containing an indication of the successful authentication along with the users' IP configuration. The

AAA-L will then use the IP configuration information to signal the local DHCP server to provide that particular client with its home network configuration. The local access router is also instructed to tunnel all traffic from that node to its home network.

A similar handling takes place if the user is accessing a PPP based connection used by e.g. CDMA2000 with the difference that the IP configuration allocation is supported by the PPP protocol. Key benefits of using this method is that on the mobility management side the traffic overhead in the access network is reduced since tunneling only takes place between access routers. Also the mobile node stays unmodified and is only required to support standardized protocols like IEEE 801.1x etc. From the AAA perspective the key benefit is that, since the AAA handling is uniform regardless of access technology, a single AAA infrastructure can be used for any set of networks. This is a very important aspect when it comes to supporting inter technology and inter provider roaming.

However, for this type of architecture to be implemented in a larger scale there is a need for a structured organization between networks and providers. For example, it is not feasible to establish a trust relationship between all AAA servers that will participate in the roaming infrastructure. The common way to solve this issue is to create a hierarchical tree infrastructure similar to the one used by the DNS system.

When deploying systems in such a large scale, an obvious question is whether the system scales properly for the number of users it might have to handle. The remainder of this paper will discuss this more issue with the focus on the AAA infrastructure. The purpose is to carry out a step towards investigating whether this type of mobility support is feasible for implementation in a large scale network.

If the AAA infrastructure fails to deliver the required level of service the whole network will suffer, regardless of the mobility management protocol performance. AAA related tasks such as user authentication can cause a significant delay when changing points of attachment. When discussing things such as handover latency, the AAA related tasks are unfortunately overseen or in some cases considered a static delay that is hard to overcome. In the proposed system architecture the whole connection procedure was taken into account and since the authentication steps required to achieve the desired security level proved to be the most resource consuming this was chosen as a focus area and the motivation for this paper.

III. AAA SERVER PERFORMANCE

AAA servers exist in a variety of implementations supporting one or more protocols. Common AAA protocols include RADIUS [5], DIAMETER [8], and TACACS+ [9]. When it comes to inter-operability and the degree to which implementations exist in currently available hardware, RADIUS is the most common protocol for AAA handling in access networks. IEEE 802.1x is used in combination with

RADIUS for most Ethernet based connections like IEEE 802.3, 802.11, and 802.16. For Wireless Wide Area Networks (WWAN) connections like UMTS or CDMA the PPP protocol is used which is actually the protocol that RADIUS was originally designed for. The RADIUS communication is typically carried out between the access network switch or access point and an AAA server so there is no need for an implementation of the protocol in an end user station.

The purpose of the AAA server is basically threefold, namely to Authenticate the user, which means to ensure that the user actually is who he or she claims to be. Secondly the purpose is to Authorize the use of a set of services that the user is entitled to access at the time. And finally to provide Accounting which basically means logging the consumption of a service. This information can be used to keep track of the service usage in order to provide a basis for billing or for network management purposes. This paper will focus on the Authentication since it is typically the most resource consuming part [10].

The authentication phase is carried out in quite different ways depending on the access network technology being used. In the evaluation case we choose to support two very different access networks namely WiFi and CDMA2000. The WiFi network supports WPA2 Enterprise with PEAP [11] (implementations exist in most common operating systems) and the CDMA2000 network supports PPP and CHAP authentication.

Figure 2. PEAP signaling

Figure 3. PPP + CHAP signaling

Figure 2 and 3 show the AAA signaling taking place for the WiFi and CDMA2000 networks authentication respectively. The actual number of PEAP messages may vary depending on implementation and configuration. This particular signaling refers to a Microsoft Windows XP client using the supplicant provided with Service Pack 2 and a FreeRadius [12] AAA server with default configuration.

From this, it can be seen that the PPP + CHAP authentication process involves 2 round-trips to complete while the typical PEAP authentication needs as much as 11 roundtrips to complete. The main reason for this is the need for increased data protection since credential details are sent over a shared medium in the WiFi case. After an initial endpoint validation step, a TLS [13] tunnel is established between the mobile host and the AAA-H in order to secure the end-to-end communication.

There are mainly three factors that may affect the AAA system performance. First, the network transit time is often the most significant reason for high authentication delays, especially if the AAA server is present at a remote location. Secondly, TLS tunnel establishment requires the generation of cryptographic information which requires a relatively large amount of CPU resources. Finally, the performance of the user information database may affect the overall performance since it can hold hundreds of thousands of user profiles that will have to be accessed randomly.

$$I = \frac{1}{(1 - P) + \frac{P}{S}} \quad (1)$$

According to Amdahls law [14] (Equation 1) an impact of a system upgrade on the total system performance may be estimated by separating the operations carried out by the system into two parts, namely parallelizable (P) and non-parallelizable (S) operations. Given that the network link to the AAA server is not saturated the network transit time is an example of a non-parallelizable part since adding more links will not speed up the process.

Cryptographic calculations and database operations on the other hand are carried out within the AAA server and may be speeded up by adding more processors and memory and can therefore be considered parallelizable.

When it comes to network delay there is a physical limit as to how fast 11 roundtrips can be done to a remote location that is hard to circumvent without reducing the number of roundtrips. A reduction of roundtrips will require a need for a modified AAA protocol.

Looking at CPU load, since the most part of the processing work for the AAA server involves cryptographic operations an AAA servers performance from that perspective is roughly proportional to the number of RSA key sign calculations that the CPU is able to carry out during a specific time period [12]. Furthermore, the user database lookup may, depending on the type of and size of the database cause a performance degradations.

The main factors at the server side that have an impact on the AAA mechanism performance and scalability can be formalized by the following formulae:

$$\bar{T}_{AAA} = n_{RTT} \times \bar{T}_{RTT} + T_{CPU} + T_{DB} \quad (2)$$

$$f_{auth} = \frac{n_{auth}}{s} \quad (3)$$

$$BW = \frac{\sum h_{pkt}}{\bar{T}_{AAA} - \bar{T}_{RTT}} \times f_{auth} \quad (4)$$

$$M = M_{Alloc} \times (\bar{T}_{AAA} - \bar{T}_{RTT}) \times f_{auth} \quad (5)$$

$$CPU = \max_{f \in R} \left(\frac{R_{RSA}}{f_{auth}}, \frac{R_{DB}}{f_{auth}} \right) \quad (6)$$

Given the signaling described in figure 2 as a worst case (only WiFi users), Equation 2 states the total AAA completion time as n_{RTT} RTTs plus the total time consumed by the AAA server CPU and by the database lookup. The value of n_{RTT} depends on the configuration and PEAP type being used. Equation 3 gives the frequency at which an AAA server is invoked where n_{auth} is the number of authentications during s seconds that the server may experience. Equation 4 expresses the network bandwidth need by the server during operation where h_{pkt} is the packet size. The reason for deducting one RTT from the divisor in Equation 4 is that from a server perspective the AAA session duration is one RTT shorter (see Figure 1).

Furthermore, the server allocates a certain amount of memory, M_{Alloc} during each authentication session and the overall memory need is calculated by Equation 5. Finally, the CPU utilization is calculated by Equation 6 as the maximum value of the RSA calculation rate, R_{RSA} and the database lookup rate, R_{DB} for each CPU. If the CPU utilization is less than 1 the CPU is sufficient to handle the wanted number of users.

IV. EXPERIMENTAL RESULTS

In order to determine the performance of a typical AAA system, a test bed and an experimental environment was created with one low-end and one high-end system. The two example servers evaluated are an Intel Pentium 4, 1.4 GHz and 1 GB RAM which is compared to a server equipped with dual Intel Xeon E5405 Quad Core CPUs and 32 GB RAM. Both systems are running 32-bit Fedora Core 12 operating system, FreeRadius 2.1.8 and MySQL 5.1.41 with the same configuration.

Figure 4. RSA key calculations

Figure 4 shows the amount of RSA key calculations that two example servers can carry out per second for 512, 1024 and 2048 bit key lengths. The results are achieved using a benchmark function of the openssl [15] implementation. The openssl library is the same one as used by the FreeRadius server for cryptographic functions. Results show that even though there is a significant difference between the two systems the Intel Pentium 4 supports >550 512 bit signs per second which is sufficient in most smaller implementations.

Figure 5. MySQL database lookups

A common user database for a larger AAA system is based

on a SQL database such as MySQL [16]. Figure 5 shows benchmarking results performed on a simple user database consisting of three fields with fixed length, namely username, password and IP address. Results show that the server performance suffers a lot when the user database increases in size. The low-end system was only able to make 7 lookups per second when the database contained 100,000 entries.

TABLE I
TYPICAL VALUES

T_{RTT}	n_{RTT}	N_{pkt}	R_{RSA}	R_{DB}	h_{pkt}	M_{Alloc}
0.02s	11	11	75,470 op/s	55,252 op/s	340 bytes	248 kbytes

Table 1 shows average measured values for a WiFi network where the AAA server is located on the same subnet. By feeding these values into equations 2-6 we are able to determine the performance that can be expected of this system. Equations 2 and 3 are using the average RTT in order to determine the total AAA completion time. The RTT however is a dynamic value that depends on a number of factors along the way and is likely to vary between each instance. In an IP network the observed RTT for any destination has proven to be fairly Poisson distributed [17]. Using this fact, we can create a 95% confidence interval within which the observed RTTs are likely to fall. Using this method we can strengthen the conclusion by saying it will be true for 95% or more of the cases. The confidence interval for a Poisson distribution is created by the following formulae where α is the confidence interval limit and d is the expected value.

$$LL = \frac{\left(\chi_{\frac{\alpha}{2}, 2d}^2 \right)}{2} \quad (7)$$

$$UL = \frac{\left(\chi_{1-\frac{\alpha}{2}, 2(d+1)}^2 \right)}{2} \quad (8)$$

Figure 6. RTT confidence interval

Figure 7. Auth/s based on bandwidth

Figure 8. Auth/s based on server memory

From the above results we can see that the main bottleneck when it comes to AAA performance is the server computational power when there are a high proportion of WiFi users and a strong encryption scheme is applied. Also, the network link bandwidth for WiFi scenarios where the AAA server is close to the access network (low RTT) and the user database is relatively small may be an issue. When considering an AAA architecture with global roaming support the conditions change quite significantly and we can see that with higher RTT, server memory becomes an issue since each authentication takes more time and therefore will allocate memory for a longer time period. Large ISPs and companies may have a large number of users. In such cases, where the user database exceeds 10,000 entries or so, the performance of the user database becomes a critical issue as well. Optimization when it comes to database engine choice etc. and load balancing may be used in order to speed up this process.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a theoretical and experimental study on questions regarding the scalability of an AAA system for supporting global roaming in a heterogeneous access network environment. A typical reference system was created and studied in terms of performance in many aspects. WiFi networks supporting WPA2 were identified as the most demanding application, AAA performance wise, among commonly used technologies. Only in WiFi networks with strong TLS encryption was the main AAA performance bottleneck identified as the AAA server CPU performance. A modern server CPU is able to perform >6,000 auths/s which is far more than a 1 GBit/s network interface can handle. The main reason for performance issues in large AAA systems were identified as the network link to the AAA server and the performance of the user database. It can be concluded that the AAA system can be built to scale well in systems with <1,000 users using standard equipment. In larger scale systems resources may be load-balanced and optimized in order to perform satisfactorily.

Future work includes implementing the system architecture described in Section 2 in a publicly available wireless network that will support seamless IP mobility between WiFi and CDMA2000 connected to a common AAA backend. Also, plans exist to implement support for sensors and other M2M devices into the same architecture that will also utilize the AAA system.

REFERENCES

- [1] C. Perkins (ed.), IP Mobility Support for IPv4, IETF, RFC 3344, August 2002
- [2] J. Rosenberg (ed.), SIP: Session Initiation Protocol, IETF, RFC 3261, June 2002
- [3] K. Andersson, D. Granlund, M. Elkotob, and C. Åhlund, Bandwidth efficient mobility management for heterogeneous wireless networks, In proceedings of the 7th annual IEEE Consumer Communications and Networking Conference (CCNC 2010), Las Vegas, NV, USA, January 2010
- [4] D. Granlund, K. Andersson, M. Elkotob, and C. Åhlund, A uniform AAA handling scheme for heterogeneous networking environments, In Proceedings of the 34th IEEE Conference on Local Computer Networks (LCN'09), Zürich, Switzerland, October 2009
- [5] C. Rigney (ed.), Remote Authentication Dial In User Service (RADIUS), IETF, RFC 2865, June 2000
- [6] WF Alliance, The State of Wi-Fi® Security: Wi-Fi CERTIFIED™ WPA2® Delivers Advanced Security to Homes, Enterprises and Mobile Devices, WiFi Alliance White Paper, 2009
- [7] EduRoam, www.eduroam.org
- [8] P. Calhoun (ed.), Diameter Base Protocol, IETF, RFC 3588, September 2003
- [9] D. Carrel, Lol Grant, Cisco Systems, The TACACS+ Protocol, IETF, Internet draft, draft-grant-tacacs-02.txt, January 1997
- [10] A. Hess and G. Schäfer, Performance Evaluation of AAA / Mobile IP Authentication, In Proceedings of the 2nd Polish-German Teletraffic Symposium (PGTG'02) Gdansk, Poland, September 2002
- [11] V. Kamath and M. Wodrich, Microsoft's PEAP version 0 (Implementation in Windows XP SP1), Internet Draft, draft-kamathppext-peapv0-00.txt, October 2002
- [12] FreeRadius, www.freeradius.org
- [13] T. Dierks, The TLS protocol, IETF, RFC 2246, January 1999
- [14] G. M. Amdahl, Validity of the Single-Processor Approach to Achieving Large Scale Computing Capabilities, AFIPS Conference Proceedings, pp. 483-485, April 1967
- [15] OpenSSL, www.openssl.org
- [16] MySQL, www.mysql.com
- [17] D. L. Mills, Internet Delay Experiments, IETF, RFC 889, December 1983