Evaluating and Improving Firewalls for IP-Telephony Environments

Utz Roedig[1], Ralf Ackermann[1], Ralf Steinmetz[1,2]
1 - Darmstadt University of Technology - Industrial Process and System Communications (KOM)
Merckstr. 25 - 64283 Darmstadt, Germany
2 - German National Research Center for Information Technology - GMD IPSI
Dolivo-Str. 15 - 64293 Darmstadt, Germany
{Utz.Roedig, Ralf.Ackermann, Ralf.Steinmetz}@KOM.tu-darmstadt.de

Abstract -- **Firewalls are a well established security mechanism for providing access control and auditing at the borders between different administrative network domains. Their basic architecture, techniques and operation modes did not change fundamentally during the last years.**

On the other side new challenges emerge rapidly when new innovative application domains have to be supported. IP-Telephony applications are considered to have a huge economic potential in the near future. For their widespread acceptance and thereby their economic success they must cope with established security policies. Existing firewalls face immense problems here, if they - as it still happens quite often - try to handle the new challenges in a way they did with "traditional applications". As we will show in this paper, IP-Telephony applications differ from those in many aspects, which makes such an approach quite inadequate.

After identifying and characterizing the problems we therefore describe and evaluate a more appropriate approach. The feasibility of our architecture will be shown. It forms the basis of a prototype implementation, that we are currently working on.

I. Introduction

A. IP-Telephony

IP-Telephony is used to establish a conversation comparable to a classic telephone call using an IP infrastructure. Typical applications and scenarios are currently based on different protocol suites. At the moment there are two main approaches - the H.323 [1] protocol family and the Session Initiation Protocol SIP [2] with a changing distribution and relevance. Though today, a high percentage of applications and scenarios is still H.323 based (and we will therefore initially focus on it), it is supposed that in the near future the use of the SIP protocol may increase [3]. Both protocol types will even be usable together with appropriate gateways.

B. Firewalls

Within a global networked environment, security aspects have become more and more important and access control at network borders is considered essential. Therefore, most organizations replaced their simple internet routers by firewalls.

These firewalls consist of packet filters, "stateful filters", proxies or a combination of all these. A firewall examines all network traffic between the connected networks. Only packets that are explicitly allowed to (as specified by a security policy) are able to pass through [4],[5]. In addition to the inspection of data flows, some firewalls also hide the internal network structure of an organization. From the Internet the only visible and therefore attackable network system is the firewall. This is achieved by the use of proxy functionality or a Network Address Translation (NAT) mechanism.

To perform its observation tasks the firewall components (filters, stateful filters, proxies) need to interoperate with a special component for the services (e.g. IP-Telephony) they want to support. We refer to this component as a parser. Based on the analysis of the traffic, the firewall decides whether packets may be passed through. A parser may also interact with NAT or proxy components since it extracts the information that can be modified or used.

II. Problem Domain

A. Multimedia Applications

The type of applications considered here are multimedia applications which use continuous (e.g. audio, video) and discrete media (control, text, meta data) data [6]. Multimedia applications significantly differ from traditional applications.

Especially

- multiple flows for one logical session,
- complex protocols and dynamic protocol behavior,
- high data rate and other QoS constraints,
- the usage of multicast mechanisms

are common features and may cause problems in a network environment which is protected by firewalls.

A comprehensive description and general approaches to deal with these characteristics can be found in [7],[8], [9],[10] and [11]. In this paper we intentionally focus on

IP-Telephony related topics.

B. Specific IP-Telephony related characteristics

Figure 1 describes a scenario in which H.323 components and firewalls are used together. It is considered to be representative for common operational areas and may slightly be adapted to individual other configurations.

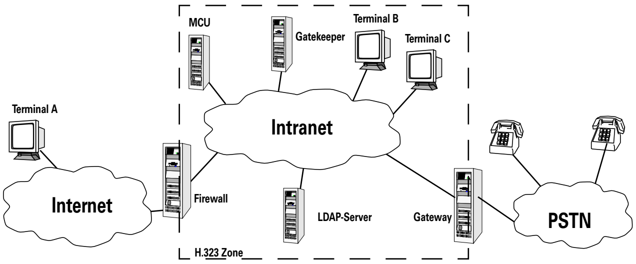

Figure 1: IP-Telephony scenario including Firewalls

The figure shows a private intranet of an organization, protected against the public Internet by a firewall. Within the intranet one or even more H.323 zones may exist. A H.323 zone consists of a gatekeeper and several optional devices such as a Multi Conference Unit (MCU), gateways and terminals.

1) Variety and complexity of communication mechanisms

The communication mechanisms used in the scenario depend on the involved components and may differ for different use cases. If only two terminals (Terminal A and C) establish a H.323 connection, the following basic order of events proceeds:

- **Q.931 (TCP) signaling:**
  Terminal A contacts Terminal C via TCP. The TCP connection is used by Q.931 to set up the call and to negotiate the parameters (e.g. ports) for the following H.245 connection.
- **H.245 (TCP) signaling:**
  Terminal A contacts Terminal C via TCP using the negotiated port. The H.245 connection is used to determine the characteristics of the following media streams (e.g. audio or video).
- **RTP/RTCP (UDP) media and control traffic:**
  Several streams may be used between the two terminals. At least 4 UDP streams are necessary to transmit audio (1 RTP and the corresponding RTCP stream in each direction). Additional streams could be used if also video has to be transmitted.

If, in the same scenario, a gatekeeper is used, the communication mechanisms differ. In this case we observe:

- **RAS registration (TCP):**
  At system start up the terminals use a TCP connection to register themselves at the gatekeeper using the Registration, Admission and Status (RAS) protocol.
- **RAS Admission Control (TCP):**
  Before the communication can be set up between both terminals, the calling terminal (Terminal A) requests a permission at the gatekeeper using the RAS protocol. If this permission is granted, the communication setup proceeds incorporating the steps (Q.931, H.245, RTP/RTCP) described above.

The communication mechanisms also change, if other devices like MCU or gateways are used.

2) Vendor specific implementations / features

Not only the use of other components within the scenario has major implications. Our experiments show, that different vendors also use different (and sometimes not interoperable) implementations, though they claim to be fully H.323 compliant.

In case the Terminal A is not a "pure" H.323 terminal but implements Microsoft Netmeeting the following extension will be used (and some firewall solutions rely on it):

- **ILS/LDAP (TCP) name / address resolution:**
  Before the communication is set up, Terminal A tries to inquire at a Internet Location Service (ILS) or Lightweight Directory Access Protocol (LDAP) server, to perform a name lookup. That way it can use a symbolic alias name to address the client (phonebook functionality). After the client has determined the destination address, it proceeds using the basic H.323 communication mechanisms. Within Microsoft NetMeeting scenarios the name lookup process is usually based on an ILS request.

The selected examples show, that the communication behavior may change significantly each time the scenario changes. As soon as the resulting control or media traffic crosses network borders, firewalls have to deal with that dynamic variety, which is not a trivial task.

3) Network Address Translation (NAT)

Another problem arises when Network Address Translation (NAT) has to be performed by the firewall. In this case the internal terminals (Terminal B and C) can not be called directly from the "outside" networks, because their address is not visible for an external terminal. This is a desired firewall function - it hides internal details and prevents internal systems from being attacked directly. It conflicts with the usual H.323 protocol flow though.

If, in our scenario (Figure 1), Terminal A wants to connect to Terminal B this could not be done directly. Terminal A has to connect to the firewall first, then Terminal A has to tell the firewall to whom it wants to talk. The firewall then has to contact Terminal B and must proxy the control / audio streams between both terminals. There exist different methods to achieve this goal.

If no gatekeeper is present in the scenario, the following method, described in [12] can be used:

- The external terminal has to be modified. There must be a configuration entryy in which the user can specify a firewall which will proxy the call.
- The calling party must connect to the remote proxy, and tell that proxy whom it wants to talk with. The H.323 setup message supports this operation mode. The destCallSignalingAddress and/or the destinationAddress (alias list) must contain the address of the proxy. The remoteExtensionAlias field should contain the information about the actual target user. The proxy must resolve the name into an IP address. This could be done by using DNS, LDAP or different protocols.
- Then, the proxy connects the target and relays the control/audio streams between both terminals.

If a Gatekeeper is present, the following method is proposed in [13]:

- The gatekeeper in the internal network has to be installed in parallel to the firewall. It has to be configured with a valid address.
- The external Terminal A has to be configured to use the gatekeeper.
- If Terminal A wants to initiate a call to Terminal B, it asks the gatekeeper for permission to call Terminal B.
- The gatekeeper responds with the address of the firewall to Terminal A.
- Terminal A calls the firewall (or the proxy within the firewall).
- The proxy consults the gatekeeper for the true destination which is Terminal B.
- The proxy then complements the call setup and relays the control/audio streams between both terminals.

In the case, that an internal terminal wants to initiate a call, the same methods can be used. In addition the firewall can try to handle the call "transparently". The internal terminal places the call to the external terminal directly, because this one has a valid address. The firewall has to monitor and remember the communication state and has to map all internal IP addresses (for internal terminals) to addresses that are valid externally (as e.g. the address of the firewall itself).

We expect both parties (calling party and called party) to be behind their own company firewall in most practical scenarios. Therefore the incoming call problem is a general and very important one. As shown above, all available solutions, to handle incoming calls in NAT environments require an interaction between the firewall/ proxy and the components that perform address resolution. The name resolution could be performed by H.323 components (e.g. a gatekeeper) or by other services (e.g. DNS, LDAP,...). Therefore a parser component within a firewall must be able to interoperate with these services.

C. *Parser related problems*

The task of traffic observation within the firewall is performed by a parser. Commonly used firewalls use static and integrated protocol parsers. These parsers are often written in a firewall specific language (e.g. INSPECT in a FIREWALL-1 [14]). Usually they are compiled in advance and then statically loaded into the firewall.

They may interact with the firewall, request data streams for analysis and reconfigure the overall system based on their inspection results. A system of this type is shown in Figure 2.

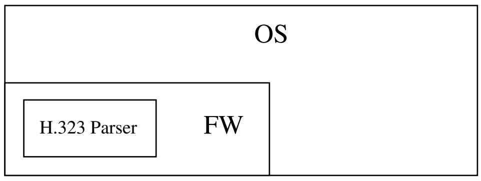

Figure 2: Integrated protocol parser

The figure shows a H.323 parser which is directly embedded (as other parsers for other protocols may be too) in the firewall (FW) itself. The firewall relies on and uses functions of the Operating System (OS) of the firewall host. IP-Telephony data streams are passed to the firewall components (e.g. by configuring OS specific sockets / packet filters) and the parser within the firewall is responsible for analyzing them. In this paper we will generally use this kind of basic schematics for explaining and comparing the differences between various architectures.

A "parser as integral part of the firewall" approach works very well with common applications, but with IP-Telephony applications it does not. The following reasons cause this fact:

1 **Different communication mechanisms:** Obviously, different parsers are necessary for every type of H.323 scenario. If the scenario is changed only slightly, the parser can often not be adjusted to the new requirements and a new parser becomes necessary. Our practical evaluation shows, that static and embedded parsers are not able to adapt to the described complex scenarios.

2 **Network Address Translation (NAT):** The parser can only communicate with the firewall, but not with other components. As shown, a connection to other components is necessary to successfully

enable the use of NAT.

III. Evaluation of Current Solutions

A "conventional" firewall/parser architecture, as shown in Figure 2, is obviously not sufficient to support IP-Telephony scenarios. This fact has been recognized by various firewall vendors and has lead to implementations cope with the problems. The first example describes the H.323 solution of the firewall market leader (80% of the market). The two other examples show dedicated solutions, which explicitly address the described problems.

A. Firewall-1

The architecture of the Firewall-1 [14] product basically corresponds to the architecture shown in Figure 2. Therefore all problems described above occur in a Firewall-1 protected network. Because the parser is static, a dedicated parser is necessary for each communication scenario. Currently two parsers are available, one for Microsoft Netmeeting and another generic one for H.323 traffic. We tested these parsers with the following results:

- The Netmeeting parser supports the direct connection between two Netmeeting (version 2 and version 3) terminals only. If one of the terminals is replaced by another product (in our experiment a H.323 compliant Innovaphone IP400 [16]), the parser does not work correctly and the intended connection setup is blocked by the firewall.
- The generic parser did not work at all. Almost no documentation is available for this parser, so the reasons for its failing could not be inspected in detail, nor could it be reconfigured correctly.
- NAT scenarios are not fully supported. There is no mechanism to handle incoming external calls in a NAT network configuration mode.

Summary:
The parser components are very static. Only some basic scenarios with standard applications could be run successfully in our experiments. Because of the missing interaction between the firewall and the parser components, inherently not all address translation scenarios can be supported.

B. Cisco MCM

The Cisco Multimedia Conference Manager (MCM) [13] provides both gatekeeper and proxy functionality. It forms an additional system which can be used to extend existing firewalls with IP-Telephony functionality. The MCM can be installed on a Cisco System (e.g. router using Cisco IOS), in parallel to or behind a firewall. Its architecture is shown in Figure 3.

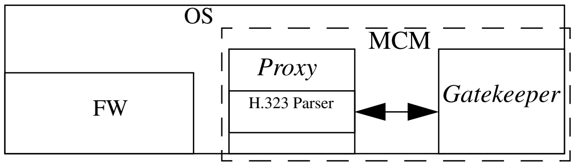

Figure 3: MCM Architecture

All IP-Telephony traffic is handled by the MCN and thereby "bypasses" the original firewall. An interaction between the firewall and the MCM is not intended. If the MCM is used parallel to the firewall, NAT scenarios can be supported. This is possible, because the MCM consists of a gatekeeper and a proxy which are able to interact.

Summary:
The approach basically addresses the NAT problem. All possible NAT scenarios could be supported. The parser within the proxy part of the MCM is also static. The parser component can not be adapted to dedicated scenarios and applications. Interaction with a gatekeeper is possible, interaction with other components like ILS or DNS is not used.

C. Phonepatch

The PhonePatch [15] component focuses on a NetMeeting scenario and works like a proxy with some additional functionality (PBX like functions, e.g. callback). PhonePatch is used in parallel to an existing firewall and is explicitly responsible for handling the IP-Telephony traffic. An interaction between the firewall and PhonePatch is not implemented.

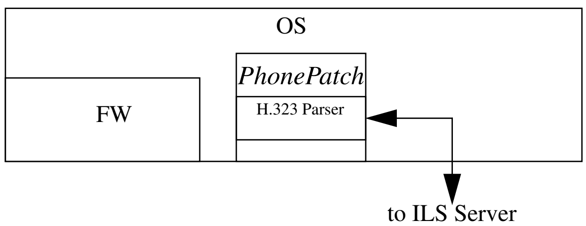

Figure 4: PhonePatch Architecture

All Internet Location Service (ILS) requests are passed through PhonePatch. This allows to examine the IP addresses transfered as part of the ILS protocol and adjust them to redirect the call from its original destination to the PhonePatch host. When data streams then arrive, the PhonePatch component directs them to the host that was mentioned in the original ILS request. This transparently fools NetMeeting applications into making a "proxy call", even though the application configuration does not have to support proxies explicitly (Netmeeting

Version 3 supports using proxies for outgoing calls now).

Summary:

This approach basically addresses scenarios using NetMeeting terminals using ILS in NAT environments.

Varying protocol scenarios and generic H.323 applications are not targeted and could not be supported in our experiments using other H.323 systems (e.g. InnovaPhone).

IV. Our new extended Approach

As we have shown, a commonly used internal firewall architecture as shown in Figure 2 is not very useful for IP-Telephony scenarios. Various vendors recognized this and implemented / proposed other architectures. These - up to now - may handle parts of the described problem domain. A general solution for all of the problems is not available yet. That is why we introduce a new parser architecture (Figure 5) which is explicitly targeted to be more general.

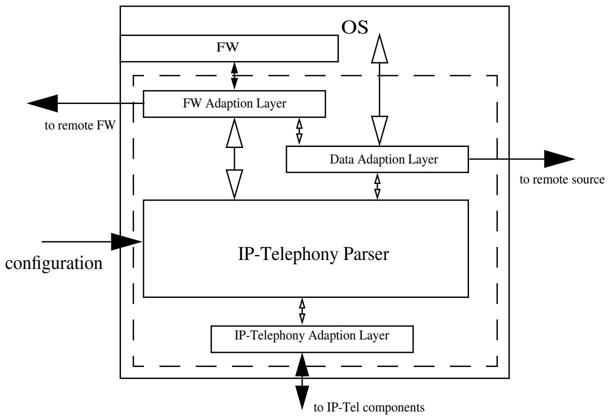

Figure 5: Proposed Alternative Architecture

We decide to place the parser outside the conventional firewall core.

- This allows the parser component to communicate with other (e.g. IP-Telephony) components. As a consequence all relevant NAT scenarios can be supported.
- Additionally the parser component can be loaded dynamically and configured separately (e.g. with an optimized / dedicated configuration language) from the firewall. This enables a general and still lightweight support for dedicated and even changing or emerging scenarios and components.

These design considerations directly influence our architectural and implementation strategy.

To be able to move the parser out of the firewall core, an interface is necessary. It allows the parser to interact with the firewall system as it did before when it was an integral part of it.

- An adaption layer is used, to allow the reuse of the parser when the firewall type/vendor is changed. The so called "Firewall Adaption Layer" is responsible for mapping the generic firewall commands generated by a specific (e.g. IP-Telephony) parser to commands that are understandable for a specific (and thereby enhanced) firewall. As an example, generic commands are used to inform the firewall, which connections are negotiated and should be passed through or redirected to a specific filter.
- We use a so called "Data Adaption Layer" which is responsible for redirecting the data streams to the parser. This layer allows to modify the internal source of the observable and modifiable data.
- We use a dedicated adaption layer for communicating with external components. In our scenario it is called "IP-Telephony Adaption Layer". The parser can generate generic requests and the adaption layer is able to map these request to the protocol language of a special component. This for example allows to map a parser request like "determine the destination address for a call to user steinmetz" to a specific DNS, LDAP and/or Gatekeeper request.

A variety of additional benefits directly results from this architecture:

- Not only can the parser be easily adapted to dedicated H.323 scenarios. It may also be changed for scenarios which use a different IP-Telephony signaling protocol. Support for SIP scenarios or heterogenous scenarios can be implemented by just modifying the IP-Telephony parser.
- As our current implementation shows, by extending the FW Adaption Layer, the parser can support different firewalls and firewall systems. The parser must not be rewritten from scratch, if ported to another system.

V. Summary

In this paper we have shown that and why the usage of firewalls leads to problems within IP-Telephony scenarios. We analyzed available firewall products and showed that they do not fully support all relevant IP-Telephony needs. To allow the unrestricted use of IP-Telephony applications within firewall environments we propose a new architecture. This one is currently evaluated as part of an experimental prototype implementation.

VI. References

[1] ITU: ITU-T Recommendation H.323, Packet-Based Multimedia Communication Systems, 1998

[2] M. Handley, H. Schulzrinne, E. Schooler, J. Rosenberg: RFC 2543 SIP: Session Initiation Protocol, March 1999

[3] Douskalis, B.: IP Telephony - The Integration of Robust VoIP Services, Prentice Hall, 2000

[4] Chapman, D. B.: Building Internet Firewalls, O'Reilly, Cambridge, 1995

[5] Cheswick, W. R.; Bellovin S. M.: Firewalls and Internet Security, Addison Wesley, 1994

[6] Steinmetz, R.; Nahrstedt, C.: Multimedia: Computing, Communications & Applications, Prentice-Hall, 1995

[7] Finlayson, R.: IP Multicast and Firewalls, Internet Draft, draft-ietf-mboned-mcast-firewall-02.txt, 1998

[8] Christoph Rensing, Utz Roedig, Ralf Ackermann, Lars Wolf, Ralf Steinmetz: VDMFA, eine verteilte dynamische Firewallarchitektur für Multimedia-Dienste, In Informatik aktuell, Kommunikation in Verteilten Systemen (KiVS), Springer, 2.-5. März 1999

[9] Utz Roedig, Ralf Ackermann und Christoph Rensing: DDFA Concept, Technical Report KOM-TR-1999-04, KOM, Dezember 1999

[10] Utz Roedig / Ralf Ackermann: Firewalls and their Impact on Multimedia Systems, Multimedia Computing and Networking 2000, January 2000, Panel Discussion "Security Firewalls and their Impact on Multimedia Systems"

[11] Ellermann, U., Benecke, C.: Parallele Firewalls - skalierbare Lösungen für Hochgeschwindigkeitsnetze, veröffentlicht in: DFN-CERT Workshop Sicherheit in vernetzten Systemen, Hamburg, 1998

[12] Intel: http://support.intel.com/support/videophone/trial21/H323_WPR.HTM

[13] Cisco: MCM, http://www.cisco.com/univercd/cc/td/doc/product/software/ios113ed/113na/1137na/mcm_cfg.htm

[14] Marcus Goncalves Steven Brown: Checkpoint Firewall 1, Administration Guide, McGraw-Hill, 1999

[15] PhonePatch: http://www.phonepatch.com

[16] InnovaPhone: http://www.innovaphone.com