Denial of Service Issues in Voice Over IP Networks

Jason Reid, Andrew Clark, Juan M González Nieto, Jason Smith
Information Security Research Centre, Queensland University of Technology
GPO Box 2434, Brisbane Q 4001, Australia
{*jf.reid,a.clark,j.gonzaleznieto,j4.smith*}*@qut.edu.au*

Kapali Viswanathan
Society for Electronic Transactions and Security
21 Mangadu Swamy Street, Nungambakkam, Chennai - 600 034, India
Kapali@sets.org.in

Keywords: Denial of service, voice over IP, H.323, security.

Abstract: In this paper we investigate denial of service (DoS) vulnerabilities in Voice over IP (VoIP) systems, focusing on the ITU-T H.323 family of protocols. We provide a simple characterisation of DoS attacks that allows us to readily identify DoS issues in H.323 protocols. We also discuss network layer DoS vulnerabilities that affect VoIP systems. A number of improvements and further research directions are proposed.

1 INTRODUCTION

The telephony service is an integral part of the operation of the vast majority of organisations. Providing and operating highly reliable systems is a significant engineering challenge requiring skilled design, critical component redundancy and a highly available maintenance capability. Ensuring the availability of the telephony service when voice and data are shared over IP networks and in the face of malicious adversaries makes this challenge all the more demanding.

The distinct nature of threats to availability presented by Denial of Service (DoS) attacks have been known for some time (Needham, 1994). It is important to note that the goal of a DoS attack is not to gain unauthorised access to a machine or data, but to prevent legitimate users accessing a service. DoS prevention continues to be an active area of research, especially in the context of Internet protocols (Aura, 2000; Schuba et al., 1997). There is, however, a clear lack of DoS literature specific to IP telephony.

The main contribution of this paper is the identification and analysis of vulnerabilities in Voice over IP (VoIP) systems that may result in denial of service when maliciously exploited by individuals. We focus our attention on the ITU-T H.323 family of protocols for VoIP. Some familiarity with the standards is assumed from the reader; in particular Recommendations H.323 (ITU-T, 2003c), H.225 (ITU-T, 2003a), H.245 (ITU-T, 2003b) and H.235 (ITU-T, 2003d), as well as IPv4 protocols.

Different classifications of DoS attacks can be found in the literature (e.g. (Aura, 2000; Millen, 1992; Center, 1997; Leiwo et al., 2000)), however they are not entirely suitable for our purpose of identifying DoS vulnerabilities in VoIP protocols. In Section 2 we present a characterisation of host behaviours that may indicate DoS vulnerabilities. This characterisation leads to a simple classification of DoS attacks. In Section 3, we discuss denial of service threats to H.323 VoIP systems that exploit IP or network layer vulnerabilities. Sections 4 and 5 investigate vulnerabilities that relate to specific H.323 protocol messages, with and without the optional security services defined in Recommendation H.235. For each of the most commonly used messages we classify their DoS behaviour according to the criteria described in Section 2. We discuss the effectiveness of the cryptographic authentication mechanisms defined in H.235 in mitigating DoS threats. In Section 6, we suggest research directions that may improve H.235 authentication mechanisms. Throughout the paper we provide various recommendations to assist in alleviating the identified DoS issues. To the best of our knowledge, this paper presents the first comprehensive study of DoS vulnerabilities in H.323 based VoIP systems.

2 CLASSIFICATION OF DoS ATTACKS

A general way of distinguishing between different types of DoS attacks is to separate them into *flooding*

and *non-flooding* (or logical) attacks. A logical attack results in a denial of service through exploiting a vulnerability in the logic or syntax of processing. A process is forced or induced to perform an action that is inconsistent with the correct operation of the system. To illustrate, consider a logical attack example from the H.323 domain; an attacker sends forged H.225 RAS unregistration request messages (URQ) for targeted endpoints to the Gatekeeper. Once they are unregistered, the endpoints cannot make or receive calls. If the protocol design allows the gatekeeper to act on unauthenticated unregistration requests, a logical attack vulnerability exists since an attacker can easily forge them.

Cryptography provides useful authentication and integrity techniques to avoid logical attacks. It is theoretically possible to eliminate denial of service vulnerabilities based on logical attacks through appropriate protocol design and careful implementation. However a design that is immune from logical attacks may be more susceptible to flooding attacks. The inter-relationship exists because the protections that ensure that a process will not act on a forged or altered message require the process to perform calculations to assess message integrity. The calculations require additional time and effort. By sending the process bogus messages, the attacker consumes finite processing resources of the target with a minimal commitment of its own.

There is an asymmetry between the minimal effort required of the attacker to commit the receiving process to a much greater amount of work. If the attacker can access sufficient bandwidth to send a large number of unauthentic messages, the process's capacity to authenticate them can be overwhelmed to the point where legitimate requests cannot be handled in a timely fashion. Denial of service results. Once again, using the example of a H.323 gatekeeper, the attacker could send a constant, high volume stream of forged H.225 RAS admission request (ARQ) messages. These forged messages include bogus integrity protection data. They will fail the integrity check and be discarded. Nonetheless, legitimate requests for admission will be swamped in the flood of forged requests.

To avoid this problem the process must be able to authenticate messages at the highest possible rate that they could arrive, i.e. the message processing capacity must be matched to the message carrying capacity (bandwidth) of the connection. With the growth in network bandwidth currently outstripping the growth in processing power, this is an increasingly difficult balancing act. It also results in the *over-engineering* of processing capacity, a potentially inefficient allocation of resources, since the message handling capability may far exceed the requirements of legitimate traffic.

The overhead of cryptographic message integrity protection only exacerbates the imbalance. There is no simple solution to this issue. Logical attacks that exploit unauthenticated messages are powerful and highly effective. Flooding attacks that consume all processing and storage resources are powerful and highly effective. There are however some promising general approaches to address this apparent conflict, which we discuss in Section 6.

We can further classify DoS attacks based on the behaviour of hosts in *response* to incoming protocol messages (*requests*) that makes attacks feasible. Our approach, contrary to others (Aura, 2000; Millen, 1992; Center, 1997; Leiwo et al., 2000), is thus host centric, avoids modelling the unpredictable nature of attackers, and allows us to readily identify potential DoS vulnerabilities in protocols. The attacks with an intent to flood will be classified under Class F and that of non-flooding attacks under Class NF.

2.1 Flooding Attacks

The behaviour leading to flooding attacks can be classified based on the amount of computation required for the response computation. The classification is as follows:

- **F.C1**: the host performs response computations for unauthenticated requests;
- **F.C2**: the host performs response computations for authenticated requests;
- **F.S1**: the host accepts large request sizes;
- **F.S2**: the host sends large response sizes;

Every behaviour of the host that is of type F.C1 could be a potential denial of service vulnerability. Therefore, the response computation with behaviour type F.C1 must be as small as possible to prevent flooding attacks. Also, the number of procedures in the system that have behaviour type F.C1 must be minimal. These observations are crucial because the attacker will essentially remain unidentified and therefore cannot be blacklisted. In other words, it is impossible to control access to network services that possess behaviour type F.C1.

It can be seen that every host that provides some form of network service will provide at least one service with behaviour type F.C1. Consider the process that authenticates the requests. The requests to the authentication process are essentially unauthenticated. Only the responses of the authentication process may be authenticated. Therefore the authenticating behaviour is of type F.C1. Clearly the authentication procedure that authenticates the requests must perform small response computations.

Procedures with behaviour type F.C2 will also be susceptible to flooding attacks. But, due to the as-

sumption that such procedures respond only to authenticated requests, it may be possible to protect the host from its protocol peer. Note that the behaviour type of the procedure for authenticating requests is F.C1. So, care must be taken to design a request authenticating procedure that performs small response computations.

Procedures with behaviour type (F.S1, F.C1) must be avoided because large request sizes require more computation than small request sizes. Procedures with behaviour type (F.S2, F.C1) must also be avoided because such procedures can be used as amplifiers to attack other hosts on the network.

2.2 Non-Flooding Attacks

The behaviour leading to non-flooding attacks can be classified based on the decision making behaviour of the host's response computation. Attackers employ non-flooding attacks to force the victim host to make incorrect decisions. The following are the classification of behaviour that may lead to non-flooding attacks:

- **NF.UA**: the response computation of the host makes crucial decisions when an unauthenticated request arrives;
- **NF.A**: the response computation of the host makes crucial decisions only on the arrival of authenticated requests;

Behaviour NF.UA poses the biggest risk of denial of service attacks. To mitigate the threat of DoS due to NF.UA behaviour authentication mechanisms can be applied. Even then, denial of service may occur when the authenticated peer misbehaves. Examples of behaviour type NF.UA include the H.225.0 RAS Unregistration (URQ) request and the H.225.0 Q.931 Release Complete message.

3 NETWORK LEVEL VUNERABILITIES FOR H.323 ENTITIES

In this section we discuss denial of service threats to H.323 VoIP systems that exploit network layer vulnerabilities. The focus is on the impact of IP denial of service attacks on H.323 system availability. Table 1 present a partial list of known DoS vulnerabilities in IPv4. We do not exhaustively review the details of each attack.

Each component of an H.323 VoIP system (terminals, multipoint controller units, gateways and gatekeepers) uses TCP and UDP protocols with both fixed and dynamically assigned port numbers. Since an H.323 VoIP system employs the full suite of IP protocols, it inherits the complete set of IP vulnerabilities. The more important vulnerabilities are now summarised.

Protocol	Message	Classification
IP	Fragment Datagram	F.C1
ICMP	Destination Unreachable	F.C1
	Time Exceeded	NF.UA
	Source Quench	NF.UA
	Redirect	F.C1, NF.UA
	Echo	F.C1
TCP	SYN	F.C1
	RST / FIN	NF.UA
RIP	Responses	NF.UA
CDP	Neighbor announcements	F.C1
DNS	Name server queries	F.C1, F.S1
	Name server responses	NF.UA

Table 1: DoS vulnerability classification of IPv4.

- **ICMP Source Quench:** can be used to degrade the quality of a media channel by effectively reducing its bandwidth. The attacker masquerades as a router and sends ICMP source quench messages to an upstream router on the media stream path to reduce the packet flow rate coming from the targeted router.
- **ICMP Redirect:** messages can be sent by an attacker to dynamically alter routing tables in hosts. Since the source is not authenticated, redirect messages can be used in a flooding attack if the update process is computationally resource intensive, as is the case on a number of Windows platforms.
- **TCP SYN Flood:** the SYN flood (Bernstein, 1996) is a powerful attack that can be directed at a number of listening ports on H.323 devices. It is implemented in the majority of denial of service attack tools. The attack exploits the fact that the TCP protocol allocates memory to store protocol state on the basis of unauthenticated requests.
- **TCP RST and FIN:** because the source of TCP packets is not authenticated, an attacker can send RST and FIN packets to terminate TCP connections (Harris and Hunt, 1999). H.225.0 call control and H.245 media control channels could be targeted. Knowledge of TCP sequence numbers is required. An attacker can monitor a TCP connection to access the sequence number via: an ARP poisoning attack in the case of switched Ethernet; through an interface device running in promiscuous mode in the case of broadcast Ethernet; or through compromising an intermediate router.

It is difficult to say whether in general terms, appli-

cation or network layer denial of service vulnerabilities pose a greater threat. While a general ranking is not possible due to individual variations in the local threat environment, it is clear that IP-based denial of service attacks present a high level of risk in an H.323 context for a number of reasons:

- the proliferation and broad availability of sophisticated, user friendly IP DoS attack tools;
- as the attacks target the network layer, attackers do not need to investigate the details of the H.323 family of protocols. Consequently, the time cost in preparing an attack is low and very little specialised H.323 knowledge is required;
- the extensive use of dynamically negotiated, high-numbered protocol ports makes it difficult to apply best-practice, restrictive firewall rule sets. H.323 components are therefore, more difficult to protect;
- registered H.323 ports present an attractive packet flooding attack target as their values are fixed and published;
- the critical functions performed by the Gatekeeper make it an attractive flooding attack target. The operation of an entire zone can be affected by attacks that target the gatekeeper, e.g. attacks directed at the Gatekeeper uni-cast RAS port can block endpoint registration requests and call admission requests[1];
- the Gatekeeper is difficult to protect because its functions typically require communication with devices on remote (potentially) untrusted networks;

Many network layer vulnerabilities are addressed in the latest version of the Internet Protocol, IPv6.

3.1 Source Address Spoofing

A number of network layer attacks employ source address spoofing, either to enhance the effectiveness of the attack, or as a required element. For example, SYN, UDP and ICMP floods are more difficult to respond to and are therefore more effective when source spoofing is used, as the typical reaction technique of blocking based on source address will not work. An attack that uses forged ICMP source quench messages to reduce the throughput of time-critical media streams requires source address spoofing. The *smurf* attack[2] also requires the ability to spoof source addresses.

[1]If a gatekeeper cannot receive call admission requests, endpoints cannot receive or initiate calls.

[2]In the smurf attack, a request is sent to a broadcast address with the forged source address of the attack target. All hosts in the broadcast domain respond to the forged source address, generating a fbod of traffi c directed at the target.

Experience has shown that source address spoofing is a significant problem on the public Internet. It also presents a threat in the context of attacks confined to private IP networks. However, in private networks different mitigation strategies are possible. One such strategy involves segmenting a private network into a number of IP subnets that are connected by layer 3 devices implementing *egress* filtering. This strategy filters out packets where the network portion of the source address is spoofed. Preventing forgery of the host portion of a source IP address is a considerably more difficult problem to solve.

In any case, the elimination of source address forgery only prevents a subset of attacks. Network and application layer flooding attacks are still possible, though identification of the offender is trivial. Being able to identify the source of a traffic flood is only useful if this allows some form of response. For example, router or firewall rules can be dynamically updated to filter packets from that source. To be effective, swift detection and reaction are required. If there is no dynamically reconfigurable filter between the target and the source of the traffic flood, as is the case when attacker and attacked are in the same transport network domain, it appears that the target must defend itself.

3.2 Self Defense

If VoIP services are to be provided reliably, H.323 entities must be capable of identifying and reacting to denial of service attacks in a timely manner. The typical response strategy involves filtering the flood traffic. Filtering traffic at the physical layer offers efficiency advantages over network or application level filters as the malicious packets can be dropped before any higher layer processing resources are consumed. Using a MAC address black-list, the physical layer can drop malicious packets as quickly as they arrive.

More research is needed to investigate how intrusion detection concepts (DoS identification) could be integrated with physical layer filtering (DoS response). We briefly describe this approach in the context of a UDP flood. Assume that a number of hosts are directing a UDP flood attack at listening UDP ports on a Gatekeeper that resides on the same segment. An IDS application detects the flood traffic, noting the source IP addresses. Using a transport layer API, the IDS places a blocking request for those IP addresses. The transport layer resolves the IP to MAC address mapping and includes the relevant MAC addresses in a black-list. The gatekeeper's transport layer stops passing IP packets arriving in Ethernet frames with with the black-listed source MAC address up to the network layer. The performance of network and application layers is no longer affected by the flood.

It is clearly important that MAC addresses cannot be spoofed [3]. Were this not the case, the defence mechanism itself could be used to launch a denial of service attack by forging the MAC address of the real target, thereby having it black-listed. Certain MAC addresses should not be blockable, including those of router interfaces, otherwise all traffic coming from the router will be blocked. Where the flood traffic is coming from a router interface, the targeted host needs a way of notifying that router that it would like traffic with a nominated source IP address and the host's destination IP address, blocked. If the source MAC of the malicious traffic is blockable, the router adds the address to its blacklist, otherwise it sends a similar request to another router, the source of the traffic.

This approach appears to offer some promising features, facilitating adaptive defense in the context of private IP networks.

4 APPLICATION LEVEL VULNERABILITIES IN H.323 WITHOUT H.235

We will first consider the scenario where the H.323 protocol stacks do not implement the H.235 standard. To be more precise, we first consider the protocol stacks that do not possess any form of request authentication during the signalling phase. Subsequently, the protocol stacks that implement the H.235 standard are investigated. It should be noted that some of the attacks described here may be dependent upon the implementor's interpretation of the H.323-related standards.

4.1 H.225.0 RAS Messages

Recommendation H.225.0 specifies the Registration, Admission and Status (RAS) signalling protocol. RAS messages are transmitted between gatekeepers and end-points over an unreliable protocol such as UDP. RAS allows a gatekeeper to manage endpoints within its zone. Table 2 summarises DoS vulnerabilities found in RAS messages. RAS protocol messages are stateless in the sense that an entity could receive any RAS message at any time. H.323 entities must maintain an internal state in relation to requests and responses (confirm or reject). This behaviour leads to almost all RAS messages being suitable candidates for flooding type attacks.

[3]To address this problem, some Ethernet switches allow each port to be associated with a fixed MAC address. Changing the MAC address to port binding requires administrative access to the switch.

RAS Message		Classification
Gatekeeper discovery	GRQ	F.C1
Registration	RRQ	F.C1
Unregistration	URQ	F.C1, NF.UA
Admission	ARQ	F.C1
Bandwidth change	BRQ	F.C1, NF.UA
Location	LRQ	F.C1
Disengage	DRQ	F.C1
Status	IRQ	F.C1, F.S1, F.S2
	IRR	F.C1, F.S2

Table 2: DoS classification of RAS messages.

When the response computation performed by the host (terminal or the gatekeeper) is *large*, then the particular response computation (and thereby the gatekeeper) can be overloaded by flooding the host with many request messages. Since messages are unauthenticated, it may be difficult to black-list the attacker, who is unidentified.

The messages IRQ and IRR could potentially be susceptible to DoS if the behaviour of the gatekeeper or the terminal falls under classification F.S2. If a large response is generated by a response computation with behaviour F.C1, then these messages can be used to flood other terminals in the network. That is, the gatekeeper or the terminal with behaviour (F.C1, F.S2) can be used as amplifiers to attack other hosts in the network.

The messages URQ and BRQ could also pose a denial of service risk of type NF.UA if they could be spoofed. For example, an attacker can manufacture an unregistration request with knowledge of the targeted terminal's IP address and RAS port number. If the gatekeeper acts on the request, the terminal will be unable to make or receive calls until it is re-registered.Similarly, if bandwidth request messages can be fabricated, it may be possible to cause a denial of service by an attacker causing the gatekeeper to allocate all of the available bandwidth to non-existent calls. This type of attack could also be used to send a fabricated *reject* message to an endpoint in response to a valid request message, causing the endpoint to incorrectly believe that a requested service is not available.

4.2 H.225.0 (Q.93x) Messages

The call signalling channel (H.225.0/Q.931/Q.932) messages are transmitted between endpoints and/or between endpoints and gatekeepers (depending on the routing model). They allow the establishment and tear-down of connections between endpoints. A reliable protocol such as TCP is used for the transport of H.225.0 call signalling messages. A number of the

messages are susceptible to various DoS activity, as summarised in Table 3.

H.225.0 (Q.93x) Message	Classification
`Setup`	F.C1, F.S2
`Facility`	NF.UA
`ReleaseComplete`	NF.UA
`CallProceeding`	NF.UA
`Alerting`	NF.UA
`Connect`	F.S2, NF.UA

Table 3: DoS classification of H.225.0 (Q.93x).

H.323 entities must always be prepared to receive an incoming `Setup` message (if they want to receive incoming calls!). Thus, there is the potential for flooding an entity on the port on which it listens for connections. Since an entity receiving a setup message performs a relatively large amount of processing and responds with several messages, there is also the potential for attacks of type F.S2.

As with the RAS messages, there is also the potential for spoofed messages to wreak havoc with entities communicating over the call signalling channel (vulnerabilities of classification NF.UA). For example, `ReleaseComplete` could be abused to prematurely terminate a call. A fabricated `Connect` message, sent prior to being expected, could cause the calling endpoint to initiate call control messages to which the receiver would not be prepared to respond. A similar approach might be successful for `Facility`, `CallProceeding` and `Alerting` messages.

4.3 H.245 Messages

Recommendation H.245 specifies a control protocol for managing multimedia bearer streams. An H.245 channel must be carried over reliable transport, either on an individual TCP connection or tunnelled through an H.225.0 call control channel. H.245 specifies four classes of messages:

1. **Request:** entities send request messages for specific actions. Every request message requires an immediate response message;
2. **Response:** entities receiving a request message are required to return the corresponding response message;
3. **Command:** commands initiate a mandatory action but require no response;
4. **Indication:** entities send an indication message when no specific action or response message is required.

Command messages are of the greatest interest in terms of their capacity for misuse resulting in denial of service. H.245 commands force the receiving terminal to perform an action related to the control of media streams. Where no integrity protection or source authentication services are provided, the H.245 commands have the potential for serious denial of service consequences as command messages can be forged or modified and the receiving terminal must perform the required action. The scope of service denial is limited to the disruption of calls that are in in the process of being established or currently in progress.

H.245 Command Messages	Classification
`SendTerminalCapabilitySet`	F.S2
`FlowControlCommand`	NF.UA, F.S2
`EndSessionCommand`	NF.UA
`MiscellaneousCommand`	NF.UA
`ConferenceCommand`	NF.UA
`H223MultiplexReconfiguration`	NF.UA
`NewATMVCommand`	NF.UA
`MobileMultilinkReconfigurationCommand`	NF.UA

Table 4: DoS characteristics of H.245 commands.

Table 4 lists the H.245 commands that have potential denial of service consequences when H.235 integrity protection is not implemented. NF.UA indicates a non-flooding attack based on an unauthenticated message. The following commands are of particular note:

- `SendTerminalCapability:` when sent as a `genericRequest` this command requires the receiving terminal to send its entire terminal capability set. Since the response is potentially large, this command may be used as an amplifier.
- `FlowControlCommand:` issued by one terminal to instruct another terminal to restrict the bitrate for a logical channel to the specified value, i.e. “don’t send data any faster than x”. The receiving terminal must comply. If the specified value is not compatible with the codec being used, the terminal must stop sending data on that logical channel so denial of service results when a flow control limit is set at a smaller value than the lowest bit-rate the codec supports.
- `EndSessionCommand:` instructs the other terminal to immediately cease sending H.245 messages. Denial of service results as the call is effectively terminated without an H.245 control channel.
- `MiscellaneousCommand:` the encryptionUpdate field of this command can be used to specify a new encryption key for the nominated logical channel. An attacker could forge a message instruct-

ing an encryption update, forcing one terminal to switch to a new key without the other's knowledge. The receiver will not be able to decrypt the stream correctly resulting in a denial of service.

5 APPLICATION LEVEL VULNERABILITIES IN H.323 WITH H.235

Recommendation H.235 specifies authentication and integrity security mechanisms for H.225.0 RAS and call signalling and H.245 call control messages. Additionally for media channels, it defines a confidentiality mechanism based on symmetric encryption. H.235 provides both network level and application-specific security mechanisms. At the network level, the specification states that H.323 signalling may be secured by IPSec or the Transport Level Security (TLS) protocol. At the application level, it defines specific authentication and integrity mechanism to be used within H.225.0 and H.245 protocol messages. Application level mechanisms come in two flavours: shared-secret based and public-key based. The authentication and integrity mechanism based on shared secrets employs a keyed message authentication code such as HMAC-SHA1. The public-key mechanism uses RSA-based digital signatures and certificates to authenticate the source and integrity of protocol messages. Both methods include the choice of nonces (time-stamps and sequence numbers) or random challenges to thwart replay attacks.

The most significant improvement obtained through the use of the H.235 cryptographic primitives is the reduction in the threat of vulnerabilities of the classifications F.C1 and NF.UA. These classifications involve the receiving host performing response computations on unauthenticated requests. So, by introducing authentication (as the H.235 specifications do), flooding attacks and non-flooding attacks involving unauthenticated messages can be thwarted. However, the authentication mechanism itself may lead to flooding attacks of type F.C2 if it cannot perform the authentication computations quickly enough.

H.235 recommends the use of TLS for securing the call control channel. It should be noted that the TLS handshake has a behaviour of type F.C1 regardless of whether TLS mutual authentication is performed. This has the potential to present a critical flooding attack target if the recipient cannot process the handshake requests at a sufficiently high rate. Since TLS is a computationally expensive protocol, an attacker can commit the Gatekeeper to significant processing on the basis of an unauthenticated request, a highly risk-prone design. The same analysis applies when the public-key based authentication mechanism specified by H.235 is used at the application level. Again, there is a significant risk of flooding attacks as an unauthenticated attacker can commit a target to the expenditure of signature verification.A delicate balance between robust authentication and speed of computation must be achieved. The ease with which authentication fields can be generated and validated will also have an impact on the system's susceptibility to vulnerabilities of type F.S1 and F.S2.

The provision of integrity and/or confidentiality services on the communication channels prevents attacks which involve monitoring and or data insertion. These will typically be non-flooding attacks which exploit vulnerable behaviour at the application level (for example, implementation flaws). The H.235 security services do not prevent an authenticated user from exploiting vulnerabilities due to protocol design flaws or implementation flaws. Of course, the attacker could be identified trivially in this instance.

6 IMPROVING H.235 AUTHENTICATION

There are promising avenues for improving the DoS characteristics of H.235 authentication mechanisms. As has already been indicated, the efficiency of the algorithm is a major factor when resistance to denial of service is desired. A trade-off exists between the certainty of the authentication, and the resistance to DoS activity. If strong authentication, with non-repudiation, is required then public key algorithms may need to be considered. Similarly, when the management of share secrets such as passwords is infeasible, public-key based schemes may be needed. However, in some cases a layering of weaker but faster authentication mechanisms, followed by stronger ones, may be appropriate (Meadows, 1999).

The essential difference between the authentication methods mentioned above is the cryptographic primitives which they employ to achieve authentication. In general a hash function can be considered to be faster, and more compact than a symmetric encryption algorithm, which can, in turn, be considered to be more efficient and more compact than a digital signature algorithm. Within each type of algorithm, the concrete choice of algorithm and length of the cryptographic keys results in different speeds of the primitives. For example, the hash algorithm suggested in the H.235 standard, HMAC-SHA1 is likely to be suitable in most instances. However, faster (and probably less secure) hash functions, such as MD5, may be appropriate in instances where anticipated traffic loads dictate such a requirement. The lowest level of authentication would be a simple reachability test,

whereby the verifier checks the aliveness of the prover by sending the prover a challenge which the prover is expected to return back. Cookies (Aura and Nikander, 1997) could be used here to dispense the verifier from maintaining state.

In all the authentication mechanisms defined in H.235 there is an asymmetry between the minimal effort required of the attacker to commit the receiving process to a greater amount of work. One general approach to address this asymmetry lies in increasing the resources that the attacker must commit before the receiving process commits processing or storage resources of its own (Aura et al., 2000).

7 SUMMARY

In this paper we have identified the types of host behaviour that may indicate a susceptibility to various denial of service attacks. The broad categories of flooding and non-flooding attacks allow us to classify the behaviours which lead to these types of attack. In each case, attacks are most likely when the host receiving a message performs actions without authenticating it. However authentication mechanisms must be carefully designed so as not introduce new vulnerabilities themselves.

Network level vulnerabilities present a range of serious threats to VoIP system availability. Flooding attacks are particularly potent due to the source address spoofing vulnerability in IPv4. Source spoofing in private networks can be addressed in a number of ways including careful IP network design combined with filtering, and providing trusted MAC address to IP address binding. Where this binding is present, flooding attacks may be filtered at the transport layer, potentially the most efficient point to perform filtering. Transport layer filtering combined with intrusion detection is proposed as an area of future research. This approach allows H.323 entities to be self defending against both network and application layer flooding attacks.

Importantly, this paper has also identified vulnerabilities in application level protocol messages that are exchanged between H.323 entities. We have investigated the problems associated with the weak or non-existent authentication mechanisms present in unsecured H.323 implementations. We have also described how the security mechanisms specified in the H.235 standard (optionally implemented by H.323 entities) assist in preventing DoS activities which rely on weak or non-existent authentication. These authentication mechanisms specified in H.235, however, introduce new vulnerabilities. We have discussed such vulnerabilities as well as promising research directions that may help in mitigating them.

REFERENCES

Aura, T. (2000). *Authorization and Availability: Aspects of Open Network Security*. PhD thesis, Helsinki University of Technology.

Aura, T. and Nikander, P. (1997). Stateless connections. In *Proc. International Conference on Information and Communications Security (ICICS'97), LNCS* 1334:87–97. Springer.

Aura, T., Nikander, P., and Leiwo, J. (2000). DOS-resistant authentication with client puzzles. In *Proc. Security Protocols Workshop 2000, LNCS* 2133:170–181. Springer.

Bernstein, D. (1996). Syn cookies. http://cr.yp.to/syncookies.html. Last access date: 10 June, 2004.

Center, C. C. (1997). Denial of Service Attacks. http://www.cert.org/tech_tips/denial_of_service.html. Last accessed: 10 June 2004.

Harris, B. and Hunt, R. (1999). TCP/IP security threats and attack methods. *Computer Communications*, 22:885–897. Elsevier Science.

Hussain, A., Heidemann, J., and Papadopoulos, C. (2003). A framework for classifying denial of service attacks. In *Applications, technologies, architectures, and protocols for computer communications*, pages 99–110. ACM Press.

ITU-T (2003a). Call signalling protocols and media stream packetization for packet-based multimedia communication systems. Recommendation H.225.0, ITU.

ITU-T (2003b). Control protocol for multimedia communication. Recommendation H.245, ITU.

ITU-T (2003c). Packet-based multimedia communications systems. Recommendation H.323, ITU.

ITU-T (2003d). Security and encryption for H-series (H.323 and other H.245-based) multimedia terminals. Recommendation H.235, ITU.

Leiwo, J., Nikander, P., and Aura, T. (2000). Towards network denial of service resistant protocols. In *International Information Security Conference (IFIP/SEC 2000)*.

Meadows, C. (1999). A formal framework and evaluation method for network denial of service. In *PCSFW: Proceedings of The 12th Computer Security Foundations Workshop*. IEEE Computer Society Press.

Millen, J. K. (1992). A resource allocation model for denial of service. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 137–147.

Needham, R. M. (1994). Denial of service: an example. *Communications of the ACM*, 37(11):42–46.

Schuba, C. L., Krsul, I. V., Kuhn, M. G., Spafford, E. H., Sundaram, A., and Zamboni, D. (1997). Analysis of a denial of service attack on TCP. In *Proceedings IEEE Symposium on Security and Privacy*, pages 208–223. IEEE Computer Society Press.