

A Protocol for Wide-Area Secure Networked Appliance Communication

Stan Moyer, Dave Marples, and Simon Tsang, *Telcordia Technologies, Inc.*

ABSTRACT

This article describes problems associated with remotely accessing networked appliances (e.g., from the Internet). Networked appliances are widely viewed as the “next wave” of devices on the Internet. We discuss some possible uses for networked appliances and the requirements for communicating with them. We present details of a solution to meet these communication requirements based on the IETF Session Initiation Protocol. In addition, we discuss the rationale for our approach and reasons other approaches were not adopted. An example of the use of SIP in this domain is presented to illustrate how the solution can be used. We conclude with outstanding challenges and reiterate advantages of this approach.

INTRODUCTION

Appliances are essential to modern life in an industrial society — from the vacuum cleaner for the carpet in the front room, through the microwave oven in the kitchen, to the mobile phone in your pocket. It is difficult to imagine living in a modern world where these convenience devices don't exist. Consider now the enhancement of these same devices with network support, giving them access to the myriad information and control sources of the Internet to enhance their functionality. Suddenly the microwave oven can download recipes and tune cooking cycles to perfection, the vacuum cleaner can report a tight bearing to a maintenance agency, and maybe the mobile phone can act as the key to the front door — the possibilities enabled by allowing appliances to extend beyond their own domain are limited only by the power of our imagination.

The applicability of the networked appliance (NA) is not limited to some futuristic Jetsons-style world. With current concerns over power management in California, it's easy to imagine the power industry being able to benefit from remote access to high-consumption devices; perhaps surrendering some limited control over your air conditioning might result

in a discount on your bill or an enhancement in the quality of supply, for example. We also see the home automation and control industry as being early adopters of NA technology since it maps directly into their current markets. The automotive industry is another area ideally placed to implement and employ NA technology. Automobiles today are increasingly reliant on microcomputers, with many systems such as braking, suspension, and fuel injection controlled by these devices. The ability to network and remotely monitor or control these devices will further enable advanced safety and reliability features to be incorporated into automobiles of the future.

Of course, this idealistic picture requires good, solid engineering underneath it to make it happen; one would take little comfort in a kitchen oven that could create the perfect soufflé if you knew it was possible for a malicious attacker to turn it on any time of the day or night; nor would a network burglar alarm, enhanced with network connectivity so that you can see who is approaching your home, be much use if a technically savvy burglar could bypass its security using a well-known hack. It is for these reasons that enabling the NA vision requires new and novel capabilities that do not exist today.

This article outlines some of the capabilities required to support NAs and proposes a solution based on the Session Initiation Protocol (SIP) [1] with some extensions. It goes on to give some examples of its application and concludes by outlining the issues that have not yet been addressed and motivate the need for further research in this area.

NETWORKED APPLIANCE DEFINITION

The term *networked appliance* is difficult to define since it can cover a huge range of devices, each with its own characteristics. In some ways it is easier to characterize what is not an NA rather than what is: we would not consider a Palm Pilot, with its reconfigurable user interface and flexible programmability, an NA; but a calculator — a dedicated function device with a minimal degree of configuration

— would be. Much of the philosophy of the definition of the NA is an extension of the idea of an appliance as described by Norman [2], and we define it to be:

A dedicated function consumer device containing at least one networked processor.

We accept that this is an imperfect definition, but it serves the purposes of this document. Typical examples of NAs include lamps, refrigerators, toasters, and TVs.

THE PROBLEM

The general issues of NAs revolve around the requirement to be able to communicate, at an application level, with a device within a local domain, such as a home network,¹ from outside of that domain. The device may optionally use IP and may use an arbitrary command set, the details of which are not necessarily known outside of the domain. Figure 1 illustrates this environment.

This scenario is complicated by requirements for security and the fact that a firewall device or network address translator (NAT) may protect access into the domain.

Figure 2 outlines the networked functions which may be used by an NA. The functions have both client (i.e., requesting) and server (i.e., responding) components. By assembling these components in different ways, both master-slave and peer-to-peer communication modes can be supported. For example, a slave NA may only support server-side functions, while the master NA will support client-side functions. For peer-to-peer communication, the NAs involved will support both client and server functions. Each NA function is now briefly described.

DISCOVERY FUNCTION

- Client: Sends discovery requests (e.g., requests for specific NAs or services). For example, the discovery client may be used to discover all NAs that can display text.
- Server: Receives and processes discovery requests. If the receiving IPA matches the requested attributes, the NA's discovery server responds to the discovery request.

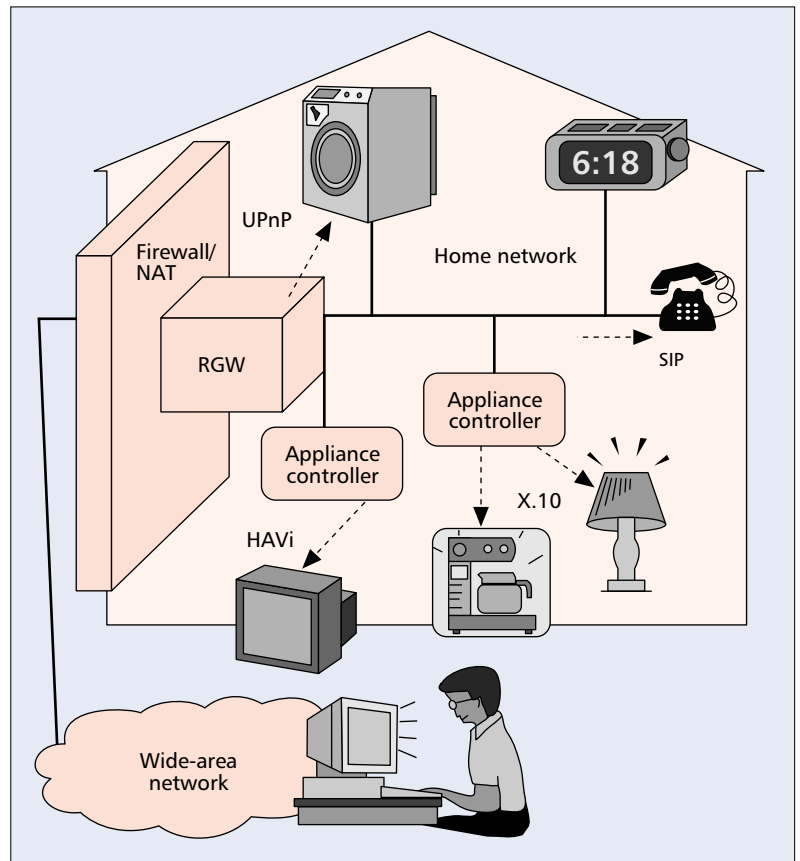
REGISTRATION FUNCTION

- Client: Sends registration information about the NA (e.g., address, name, services supported).
- Server: Receives and processes registration information. This information may then be stored (e.g., in an NA registration information database).

NA CONTROL FUNCTION

- Client: Used to control NA actuators. An example of NA control is a request to turn on a lamp or activate locks.
- Server: Receives and processes NA control

¹ Note that this work is applicable to any type of local domain (e.g., home network, vehicular network, remote enterprise network), but for the purposes of providing a simple motivating example, we will mainly discuss home networks in this article.

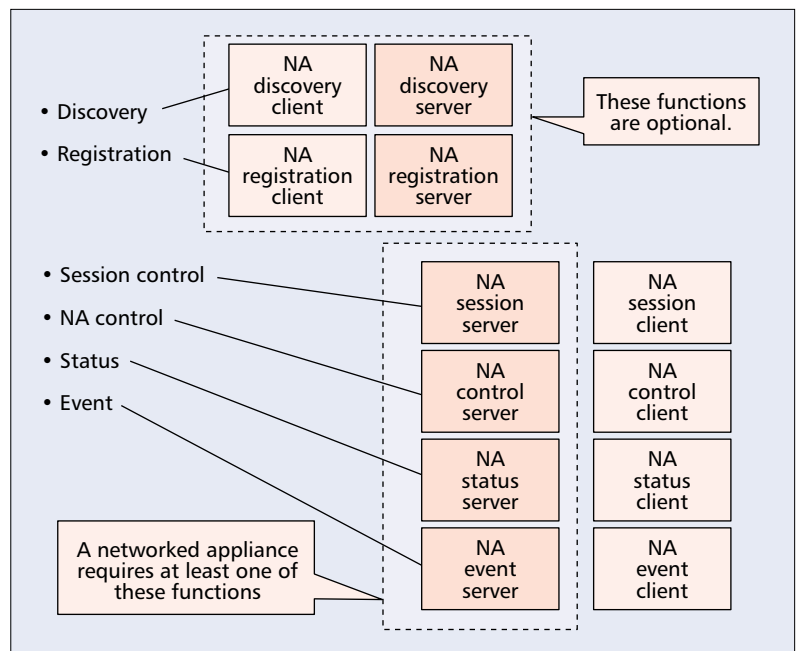


■ Figure 1. An example of home networked appliance architecture.

messages to affect its actuators. This action may require interworking the control message with the NA's native control protocol.

STATUS FUNCTION

- Client: Used to query the status of an NA sensor. For example, this could be used to



■ Figure 2. An overview of NA functions.

Additional requirements come into play when networked appliances move into environments other than their home environment. In this case the visiting appliances and their users must be explicitly authenticated and authorized before being granted access to the visited environment.

query a temperature sensor or reading on a tachometer.

- Server: Receives and processes status requests. Responds with NA sensor status information.

EVENTING FUNCTION

- Client: Used to request notification of specific events from another NA. For example, this could be used to request an event notification when the temperature sensor reading exceeds 100°, or the tachometer reading exceeds 60 mi/h.
- Server: Receives and processes event notification events. When the specified event occurs, the Eventing server will respond with a notification message.

SESSION CONTROL FUNCTION

- Client: Used to initiate a session. This facility can be used to set up media streams between two or more NAs. For example, it may be used to stream the output of a networked camera to a workstation.
- Server: Receives and processes session initiation requests.

A variety of different protocols may support these functions, but in order to provide interworking and interoperability, it is hoped that some standardization or agreement on the protocols will be reached.

Note that in this article we specifically exclude service advertising and discovery from the problem space. We believe that it will be possible to add these capabilities later based on the designs to meet the initial requirements of secure local-protocol-neutral wide-area, access.

REQUIREMENTS

The requirements for SIP for appliances are presented in [3] but an overview is provided here for completeness.

GENERAL REQUIREMENTS

It must be possible to configure access to NAs from outside of the home environment. It must be possible to interwork with different home networking technologies transparently, and NAs must be able to move within the home domain, across home domains, within the service provider's domain, and across service provider domains; and support must be provided for locating and controlling NAs in these environments. NAs should strive for auto-configuration wherever possible, and shall support usage monitoring and charging systems that may be developed for the home network.

NAMING AND ADDRESSING REQUIREMENTS

For NAs without IP capabilities, an appliance controller may be used to provide interworking between the NA and the IP network. NAs must be assigned an address in a generic format that can be used by any communicating entity in order to accommodate both IP and non-IP NAs. The naming scheme must support classification of NAs and selection between them, and it must be possible to perform searches based on these characteristics.

COMMUNICATION PROTOCOL REQUIREMENTS

The communication protocol must provide a flexible and reliable payload capability that will allow the transport of commands to, and responses from, individual NAs or classes of NAs. There must be a separation of transport and data so that new data structures can be introduced.

The protocol must support efficient messaging for control in light of the short messages that are expected in a NA system.

COMMUNICATION MODE REQUIREMENTS

In addition to control messages, the protocol must support event subscription and notification and a polling/querying mechanism. It must also be capable of setting up ongoing streaming sessions.

SECURITY CONSIDERATIONS

Security is a primary concern in these systems. Authentication, authorization, privacy, and replay protection are required in all communications, and it should be possible to check communications with devices from the wide area at different granularity levels (e.g., per session, per message, or perhaps per day). The target device name and contents of the messages must be kept private so that eavesdroppers cannot learn about what is in an individual's home.

Additional requirements come into play when NAs move into environments other than their home environment. In this case the visiting appliances and their users must be explicitly authenticated and authorized before being granted access to the visited environment.

Finally, NAs should obviously be resilient to security attacks and still be able to perform a minimum of functions correctly, even in the absence of external communications.

THE PROPOSED SOLUTION

This section describes a solution that meets the requirements identified above. First details of the solution are presented. Reasons for the selection of this approach and the rejection of others are then discussed.

SIP FOR APPLIANCES

Most of the requirements for application layer NA communication appear to be met by the Session Initiation Protocol (SIP) [1]. SIP allows abstract naming, provides end-to-end security, and can carry a flexible payload. These features make it an attractive base for a solution.

The most important message in SIP is INVITE, for it is the INVITE message that is used by one user to request another user to join a session (e.g., a phone call). SIP routes messages using a hop-by-hop routing algorithm based on rewriting message headers. A typical INVITE sequence delivers the payload (which is typically a Session Description Protocol, SDP, packet when SIP is used for voice calls) to the destination endpoint where it is actioned. The SIP security architecture enables verification that any message arriving at the destination endpoint is valid.

“Standard” SIP [1] lacks some of the capabilities needed to meet the requirements for NAs outlined above:

- Support for NA naming
- Support for encapsulating NA characteristics
- Support for communication modes other than session setup (stream-oriented communications), such as asynchronous notification and datagram operation

If these shortcomings can be addressed, SIP becomes a very practical method of communicating with NAs.

MODIFICATIONS AND EXTENSIONS TO SIP

To allow SIP to be used in the NA space it has been enhanced with the following modifications and extensions.

URL CHANGES

In SIP messages, the names that are found in the `To:` and `From:` fields are encoded as universal resource locators (URL). Current implementations support SIP and PHONE URLs. A new type of URL can be defined without changing the nature of the protocol, which allows for a more user-friendly description of the NA address inside the home. A typical address might be

`d = lamp, r = bedroom`

By base64 encoding this address (and if necessary encrypting it to avoid revealing information about the types of devices contained in the domain), it is possible to structure this as part of a SIP URL; for example:

`sip:a458fauzu3k3z@stan.home.net`

Thus, the existing SIP address structure of `<entity>@<location>` can be maintained even when extended to accommodate appliances. SIP can carry a structured NA naming scheme, but a “standard” convention for assigning names to NAs must be created. This has been outside the scope of the work so far.

A NEW EXCITATION METHOD

SIP was initially created with the model of call setup in mind to establish a relationship between two endpoints such that ongoing bearer paths can be established between them. This structure could be generalized to cater to short-lived connections if the connection establishment phase were removed and the payload generalized. The difference between the way SIP is currently used and the proposed modifications is analogous in many ways to the difference between TCP and UDP and other session/datagram protocols.

A new method, called `DO`, was defined to meet the above requirements [4]. This method, like any SIP method, can carry payloads other than SDP. Any MIME type [5] could be used as the payload, and new MIME types could easily be defined for action languages for particular classes of appliances. `DO` would carry the command that is appropriate for the target appliance, such as *turn the light on*. The command would trigger a single response, indicative of its result, which would be carried by the standard SIP response mechanisms.

NEW PAYLOAD TYPE(S)

The typical MIME payload for SIP INVITE messages is Session Description Protocol (SDP). For NAs, a payload type specific for communicating with devices is required. A new MIME type called Device Message Protocol (DMP) [6] has been defined for this purpose. It is an XML-based specification similar to that employed within the Universal Plug ‘n Play’s Device Control Protocol.

In addition, when a device registers with a proxy (via the `REGISTER` message), a description of that device needs to be carried. A Device Description Protocol (DDP) is proposed to carry this information, although the exact details are still under development. It is likely to also be XML-based and will leverage existing work in this area.

SUPPORT FOR NOTIFICATION/EVENTS

In addition to synchronous communication with NAs, a need also exists for asynchronous communications, perhaps to be notified when an alarm goes off in the home, a certain temperature is reached, or someone rings the doorbell.

The SIP event notification work [7] defines two new methods, `SUBSCRIBE` and `NOTIFY`, that can be used to achieve asynchronous communications. When these two methods are used in conjunction with the proposed URL changes and the Device Messaging Protocol MIME type, asynchronous event notification from and between networked appliances is enabled.

WHY SIP?

Even with these enhancements, the question remains of why SIP should be used to address this problem. In this section we provide some rationale for this decision.

INTEROPERABILITY

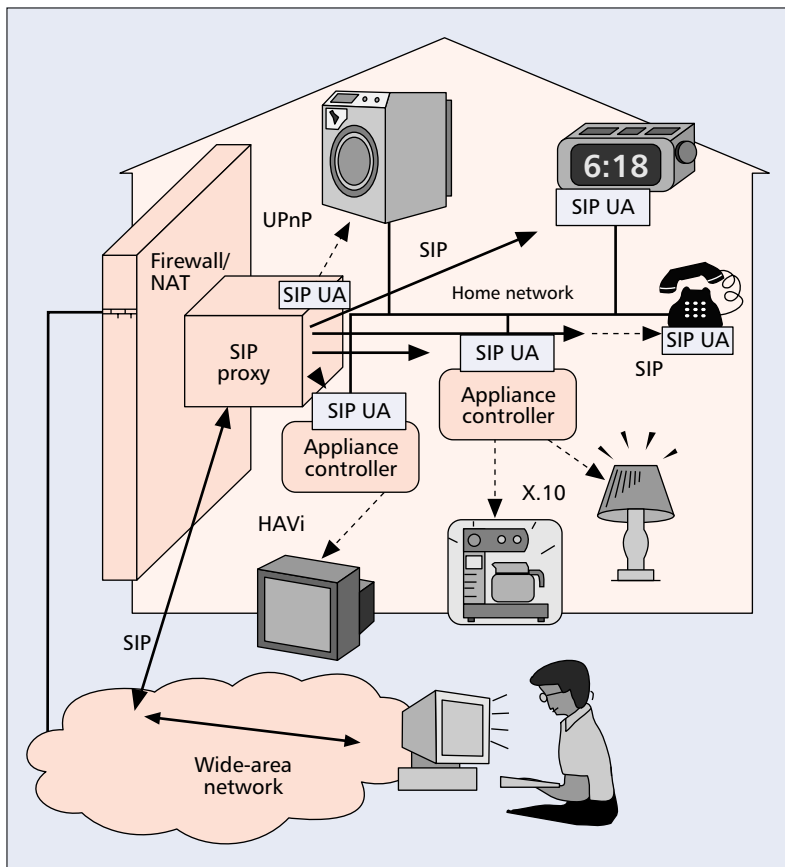
The SIP solution enables communication with devices in the local area without concern for the type of local device communication protocol being used. For example, given an architecture like that pictured in Fig. 1, there may be several local device communication protocols in use — UPnP, HAVi, X.10, and SIP. The SIP architecture assumes that an appliance controller exists to provide the necessary protocol translation/interworking to other protocols.

SIP for NAs uses SIP to communicate through the WAN to the home domain, where the SIP message is authenticated and authorized and then translated to the appropriate local device communication protocol. This is shown in Fig. 3.

SECURITY

SIP can provide both authentication and encryption. SIP for ANs utilizes these inherent capabilities to provide the security required for communication with NAs from the wide area. If all SIP messages entering the local home domain are authenticated, an authorization check can be performed before the requested operation is executed. In addition, SIP provides a means of encrypting most portions of the message, including the payload, for end-to-end privacy. Note, however, that the address is not encrypted, so if

In addition to synchronous communication with NAs, a need also exists for asynchronous communications, perhaps to be notified when an alarm goes off in the home, a certain temperature is reached, or someone rings the doorbell.



■ **Figure 3.** The use of SIP for NAs.

the appliance name and/or description are included in the address, that portion of the address must be encrypted separately, as previously described.

SCALABILITY

SIP is a very scalable protocol. It works well in both LAN and WAN environments, and it does not rely on multicast/broadcast technologies to reach a destination endpoint. SIP has a strong concept of routing that enables a packet to traverse from source to destination using intermediate existing routes, hopping from one node to another until it reaches its final destination. Additionally, SIP works over both UDP and TCP; UDP allows SIP servers to scale well. This, coupled with the fact that SIP allows various definitions of servers with incremental statefulness and thus incremental overhead, enables implementers to select the ideal combination they need based on network load and required functionality.

MOBILITY

SIP supports the concept of mobility. It is enabled through the use of REGISTER messages and SIP registrar servers. Consider an example of a user trying to remotely access their Tivo box to instruct it to record a particular program. The accessing user may not know the Tivo has been moved from the bedroom to the living room. They simply send the request to `tivo@myhome.net` and the residential proxy server in the house knows that `tivo@myhome.net` is now an alias

that points to `tivo@livingroom.myhome.net`. This happened when the Tivo was moved: it registered the new location with its local registrar. Note that the Tivo could have been moved to a different house and the command still would have worked.

EXTENSIBILITY

SIP is an extensible protocol — it allows new method/message types, new types/forms of addresses, and any kind of MIME body type. Through these capabilities, extensions were easily incorporated to enable the SIP for NAs solution to communicate with NAs in four different ways:

- Control (e.g., “turn on the coffee maker”)
- Query (e.g., “what’s the temperature in the house?”)
- Event Notification (e.g., “tell me when my fire alarm goes off”)
- Multimedia Session (e.g., “view the babysitter cam”)

Currently those are the only four types of communication modes identified for NAs and SIP supports them all. If other communication modes are required in the future, the extensible nature of SIP should easily accommodate the necessary modifications.

SIP is also flexible in the payload it carries; like e-mail, SIP carries a MIME type payload. MIME types can be defined for most types of payload [5].

SERVICE CONVERGENCE

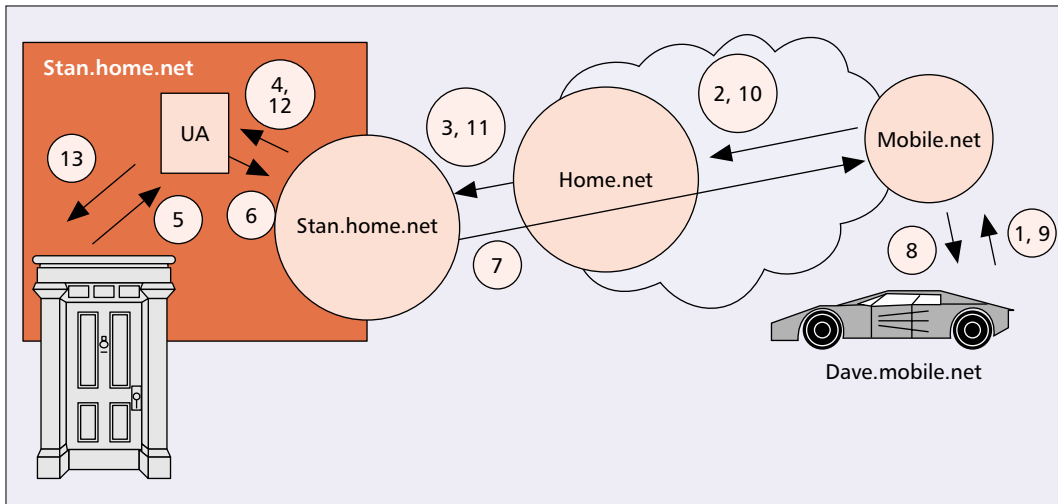
The most popular current application for SIP is Internet telephony. SIP, with extensions, is also being considered for instant messaging [8]. The same SIP infrastructure deployed for these services can also be used for NA services. The ability for a service provider to deploy a single infrastructure and offer such a wide range of services is very attractive. Only having to deploy, install, and manage one infrastructure significantly eases their administrative and operational overheads.

WHY NOT ...?

Since SIP requires some modification to enable NA communication, it is only natural to ask why a different protocol should not serve as the starting point. The following sections describe the arguments applied in rejecting possible alternatives.

HTTP

The Hypertext Transfer Protocol (HTTP) is the basis for most Web communication. An NA could be considered a Web server, with the services/applications it offers being analogous to Web pages. However, an NA could be mobile and can generate asynchronous notifications; HTTP does not provide good support for mobility and notifications. Additionally, current HTTP must run over TCP, and a TCP stack is larger and more complex than a UDP stack. This can be an issue in small devices or appliances with stringent memory and processing requirements. In addition, HTTP is inherently a client/server protocol, which does not map well to asynchronous notification scenarios.



■ **Figure 4.** Answering the front door from a car.

SMTP

The Simple Mail Transport Protocol (SMTP) seems, on first inspection, a likely candidate for NA communication since it supports an application-level addressing scheme, carries a flexible payload, and supports mobility, as exploited in e-mail forwarding. However, SMTP does not support events and media-based sessions, and it can exhibit extremely high latency, whereas NA communication frequently mandates near-real-time responses.

SNMP

The Simple Network Management Protocol (SNMP) is already popular for Internet-based network management. An NA could be viewed as a managed object, but unlike a managed object, which has a client/server relationship, NAs can be peers, they can initiate communication, and they can be mobile. Also, NAs need an application layer name and SNMP only supports network-layer addressing. Furthermore, SNMP does not support multimedia sessions.

A NEW PROTOCOL

If a new protocol were designed from scratch to solve the problem described previously, it undoubtedly would not look like SIP. It would, however, most likely share many of the same characteristics. We believe that the benefits of being able to reuse the SIP infrastructure provide sufficient motivation for not creating yet another protocol.

EXAMPLE

This section provides an example to illustrate how SIP for NAs can be used to communicate with networked appliances. The example we provide depicts how one could answer the front door from a car. In this example, Stan is riding with Dave in Dave's car and remembers that he was expecting a service person to come and fix the dishwasher; he does not have his Web phone. He asks to borrow Dave's phone and sends a message to his service provider to notify him if someone rings the doorbell.² When the service person rings the doorbell (and authenticates

him/herself with an ID badge), a message is sent to Dave's Web phone for Stan indicating that the service person is at the front door. After verifying that it is indeed a person from the right company, Stan issues a command to unlock the front door and let the person in (Fig. 4).

1. SUBSCRIBE sip:[d=door,r=front,u=starm]@home.net SIP/2.0
From: sip:stanm@dave.mobile.net
To: sip:[d=door,r=front,u=stanm]@home.net
Via: dave.mobile.net
Content-type: application/dmp
<event>ring</event>
2. SUBSCRIBE sip:[d=door,r=front,u=starm]@home.net SIP/2.0
From: sip:stanm@dave.mobile.net
To: sip:[d=door,r=front,u=starm]@home.net
Via: mobile.net
Via: dave.mobile.net
Content-type: application/dmp
<event>ring</event>
3. SUBSCRIBE sip:[d=door,r=front,u=stanm]@stan.home.net SIP/2.0
From: sip:stanm@dave.mobile.net
To: sip:[d=door,r=front,u=stanm]@home.net
Via: home.net
Via: mobile.net
Via: dave.mobile.net
Content-type: application/dmp
<event>ring</event>
4. SUBSCRIBE sip:[d=door,r=front,u=stanm]@ua.stan.home.net SIP/2.0
From: sip:stanm@dave.mobile.net
To: sip:[d=door,r=front,u=stanm]@home.net

The most popular current application for SIP is internet telephony. SIP, with extensions, is also being considered for instant messaging. The same SIP infrastructure deployed for these services can also be used for networked appliance services.

² We assume that Stan has to enter some authentication code that will be attached to the message to verify that it is Stan and not Dave who is requesting this.

SIP, with the modifications outlined in this article, provides a suitable protocol for interacting with networked appliances across the Internet. There are still outstanding issues which need to be resolved if the goal of supporting networked appliance services using SIP is to be realized.

- Via: stan.home.net
Via: home.net
Via: mobile.net
Via: dave.mobile.net
Content-type: application/dmp
<event>ring</event>
5. (Doorbell Rings! Credentials established.)
6. NOTIFY stanm@dave.mobile.net SIP/2.0
From: sip:[d=door,r=front,u=starm]
@stan.home.net
To: stanm@dave.mobile.net
Via: ua.stan.home.net
Content-type: application/dmp
<event>ring</event>
<identity>Maytag Repairman</identity>
7. NOTIFY stanm@mobile.net SIP/2.0
From: sip:[d=door,r=front,u=starm]
@stan.home.net
To: stanm@dave.mobile.net
Via: stan.home.net
Via: ua.stan.home.net
Content-type: application/dmp
<event>ring</event>
<identity>Maytag Repairman</identity>
8. NOTIFY stanm@dave.mobile.net SIP/2.0
From: sip:[d=door,r=front,u=starm]
@stan.home.net
To: stanm@dave.mobile.net
Via: mobile.net
Via: stan.home.net
Via: ua.stan.home.net
Content-type: application/dmp
<event>ring</event>
<identity>Maytag Repairman</identity>
- At this point, Stan may wish to initiate a video session with the camera over his front door if he wants to verify the identity of the person. This would be done using standard SIP (i.e., send an INVITE message to establish a multimedia session).
9. (User alerted and decides to unlock the door)
DO sip:[d=door,r=front,u=stanm]
@home.net SIP/2.0
From: sip:stan@dave.mobile.net
To: sip:[d=door,r=front,u=stanm]
@home.net
Via: dave.mobile.net
Content-type: application/dmp
<command>unlock</command>
10. DO sip:[d=door,r=front,u=starm]
@home.net SIP/2.0
From: sip:stan@dave.mobile.net
To: sip:[d=door,r=front,u=starm]
@home.net
Via: mobile.net
Via: dave.mobile.net
Content-type: application/dmp
<command>unlock</command>
11. DO sip:[d=door,r=front,u=starm]
@stan.home.net SIP/2.0
From: sip:stan@dave.mobile.net
To: sip:[d=door,r=front,u=starm]
@home.net
Via: home.net
- Via: mobile.net
Via: dave.mobile.net
Content-type: application/dmp
<command>unlock</command>
12. DO sip:[d=door,r=front,u=starm]
@ua.stan.home.net SIP/2.0
From: sip:stan@dave.mobile.net
To: sip: sip:[d=door,r=front,
u=starm]@home.net
Via: stan.home.net
Via: home.net
Via: mobile.net
Via: dave.mobile.net
Content-type: application/dmp
<command>unlock</command>
13. <Unlock!!!>

CHALLENGES

SIP, with the modifications outlined in this article, provides a suitable protocol for interacting with networked appliances across the Internet. There are still outstanding issues that need to be resolved if the goal of supporting networked appliance services using SIP is to be realized. These include those discussed here.

DEVICE DISCOVERY AND REGISTRATION

How can applications be written to control appliances if we do not know the appliance exists, what its name is, how to address the appliance, or the appliance's capabilities? It is not acceptable to assume that this information is preprogrammed since many appliances will be mobile and freely move in and out of domains. A registration capability that allows NAs to register their name, address, and capabilities and a discovery capability in the appliance that responds to queries for particular types of appliances or appliance capabilities are both required. Candidate protocols for this capability include the Service Location Protocol (SLP) [9] and Salutation [10].

SECURITY AND ACCESS MANAGEMENT

Security is a particularly important issue since the technologies described in this article will be applied to home environments with personal and private information available online. At a minimum, message authentication and encryption are required to ensure that the appliance signaling cannot be intercepted, modified, or copied. Furthermore, it is likely that multiple service providers and users may have restricted access to appliances within the home. Supporting this type of functionality requires a fine degree of access control. Existing access control methods are not sufficient, since they generally do not control multiple protocol layers.

CONTROL PROTOCOL

As described in this article, SIP provides a secure "carrier" for control messages destined for appliances. It does not define the messages themselves. The Device Messaging Protocol (DMP) has been proposed for this purpose, but it is still in the early stages. DMP must be able to accommodate different in-home networking technologies, and be backward-compatible with legacy non-IP technologies. Furthermore, the applicability of alternate protocols to DMP must be assessed.

INTEROPERABILITY AND STANDARDIZATION

In order for SIP to be applicable to the appliance space it needs to operate in a standard fashion that meets the requirements of all of the players. Otherwise, competing approaches to solving the problem will be proposed, and the home access space will become as confused as the in-home space already is. To achieve this requires the buy-in of many interested parties who are prepared to work together to ensure that the resulting standard is suitable for all users. In many ways, achieving this cross-industry agreement is more difficult than creating the approach itself.

NAMING CONVENTION

The standards to be used when naming devices in the home need to be standardized to some degree so that clashes can be avoided and new entrants into the environment are familiar with the devices already present. This naming must be generalized such that it is applicable not just to SIP-based systems, but other communications technologies too.

CONCLUSION

This article has outlined the need and an approach for secure remote control of networked appliances via the Internet. The approach uses the IETF Session Initiation Protocol with extensions to carry control messages securely across the Internet to NAs. SIP is already beginning to be widely deployed for voice over IP and instant messaging applications. Extending SIP to support NA services enables service providers to incrementally support multiple "converged" services, using the same control infrastructure (SIP-based). This provides many benefits in terms of software and equipment consolidation, and unified management and operation support systems. Users benefit from fast rollout of exciting new services and mature customer support infrastructures.

ACKNOWLEDGMENTS

This article documents work that has been done as part of the Networked Devices research project at Telcordia Technologies Inc. Many members of the Networked Devices team have contributed to the work contained in this document, and it would be inappropriate if this work was not acknowledged. Some of these people are Thanh Cheng, Ashutosh Dutta, Proveen Gurung, and Sumit Khurana. In addition we would like to thank Henning Schulzrinne and Arjun Roychowdhury for their assistance in publicizing this work outside of Telcordia and bringing it into the IETF.

REFERENCES

- [1] M. Handley *et al.*, "SIP: Session Initiation Protocol," IETF RFC 2543, Mar. 1999.
- [2] D. Norman, *The Invisible Computer*, MIT Press, Oct. 1999.
- [3] Tsang *et al.*, "Requirements for Networked Appliances: Wide Area Access, Control and Interworking," draft-tsang-appliances-reqs-01.txt, available at <http://www.arggreenhouse.com/iapp/draft-tsang-appliances-reqs-01.txt>
- [4] A. Gulbrandsen, P. Vixie, and L. Esibov, "A DNS RR for Specifying the Location of Services (DNS SRV)," RFC 2782, Feb. 2000.
- [5] N. Borenstein and N. Freed, "MIME: Multipurpose Internet Mail Extensions," RFC 1521, Sept. 1993.
- [6] S. Khurana, P. Gurung, and A. Dutta, "Device Message Protocol (DMP): An XML Based Format for Wide Area Communication with Networked Appliances," Nov. 2000, expired Internet draft; <http://search.ietf.org/internet-drafts/draft-khurana-dmp-appliances-00.txt>
- [7] A. Roach, "Event Notification in SIP," Internet draft draft-ietf-sip-events-00.txt, July 2001.
- [8] J. Rosenberg *et al.*, "SIP Extensions for Instant Messaging," Internet draft draft-ietf-simple-im-00.txt, Apr. 2001.
- [9] J. Veizades *et al.*, "Service Location Protocol," RFC 2165, June 1997.
- [10] Salutation, <http://www.salutation.org>

BIOGRAPHY

STAN MOYER [SM] (stanm@research.telcordia.com) is director of the Internet Service Infrastructure Research group in the Internet Architecture Research Laboratory in Telcordia's Applied Research, where he has been working since 1990. His current research interests include home networking and network architectures, protocols, and operations for supporting networked devices and appliances. In the past he has worked on ATM switch hardware, broadband network architectures and protocols, middleware, CORBA, Internet network and application security, Internet QoS, and voice over IP.

DAVE MARPLES (dmarples@research.telcordia.com) is a chief scientist in the Internet Architecture Research Laboratory in Telcordia's Applied Research, where he has been working since 1999. His research interests include networked appliances, digital rights management, and mobile location infrastructures. In the past he was CTO of a technology startup in the United Kingdom; prior to that he worked for GPT Ltd. in the Advanced Technology Group. He is a past Industrial Fellow of the Royal Commission for the Exhibition of 1851, and he obtained his Ph.D. from Strathclyde University in Scotland. He is honorary professor of telecommunications at Stirling University in Scotland and a member of the board of the Open Services Gateway Initiative (OSGi). He is a member of the IEE.

SIMON TSANG [M] (stsang@research.telcordia.com) is director of the Internet Service Access Research group in the Internet Architecture Research Laboratory in Telcordia's Applied Research. His current research interests include service architectures, protocols, and operation support systems for networked devices and appliances. His previous research interests included voice over IP architectures and protocols design, intelligent networks, APIs for advanced services, and the feature interaction problem. Prior to joining Telcordia he was a systems engineer in BT Laboratories. He obtained his Ph.D. in 1997 from the University of Strathclyde, Scotland. His thesis was "Behavior Modeling and Control for Feature Interaction Detection and Resolution in Intelligent Networks."

Extending SIP to support networked appliance services enables service providers to incrementally support multiple "converged" services, using the same control infrastructure (SIP-based).