1. Date of Receipt: 4 Mar. 2003

2. Registration Number: 1068

3. Applicant's Name: Dean Swift <dean@xirium.com>

4. Entry Category: Idea Award

5. Name of Entry: Xirium Metaverse

Xirium Metaverse

Distribution

The distribution of this document is highly restricted. Some names, clauses and trademarks may be removed before publication. All technical descriptions of software and protocols herein remain unpublished. This is a pre-requisite for the IP Version 6 Application Contest 2003.

Abstract

IP4 is sufficient to allow basic operations within a multi-user, distributed, graphical environment. However, IP6 is required for advanced operations. The terse and unified address space of IP6 allows all clients to be notified by servers. It also allows low bandwidth clients to securely initiate operations between high bandwidth intranets. In typical cases, cost and efficiency benefits are lost through IP6/IP4 tunnelling.

Current Practice

Proxies and firewalls have become an everyday solution to an increasingly hostile public network. Suspicious or unknown data is denied. Unfortunately, this also denies the adoption of novel protocols. Therefore, novel protocols have taken to tunnelling through existing protocols. Currently, numerous services tunnel as web connections. This is itself becoming a security risk. More subtle problems also exist.

Firewalls are used in conjunction with address translation services. The role of caches, proxies and firewalls become confused with network address translation [NAT]. This occurs because devices are typically feature rich. They typically perform more than one function. So, NATs denying transfer of data are firewalls. NATs translating awkward protocols are proxies [1]. NATs aggregating requests are caches.

The value of combined functions obscures the collapse of address space. Indeed, this is often touted as a feature. Clients access external services via one NAT. Often this is via a single address. Malicious data has little opportunity, if any, to reach a client. This is not through any strengths of gateway implementation. It is simply a lack of direct addressing. Such casual anonymity reduces pressure to secure clients - with devastating results.

It has also become common practice to establish tiers of NATs [2]. This allows a hierarchy of addresses to be established quickly and with little regard for uniqueness. Unfortunately, it also increases critical points of failure. The shortage of addresses also creates a sharp division between client and server.

Clients utilising NAT can initiate transactions with servers. However, servers cannot contact clients, simply because there is no method to address a client. In a tiered arrangement, servers may be unable to peer. This is a desirable property because it allows servers to share data directly and at the instruction of a client. However, a reciprocal ability to initiate connections may not exist. Clients and servers within NAT tiers are similarly aloof from external servers. This limitation is not obvious to clients, which have a unified address space. So, a set of servers may be visible to a given client, but not to other servers.

The lack of address space also increases the value of addresses. Servers must obtain premium addresses; visible to clients. Whereas the incremental cost of a client address is zero. Fortunately, virtual hosting allows organisations to share server addresses. This is one of the major features of HTTP/1.1 lacking from previous versions [3]. Such solutions mitigate cost. Unfortunately, such solutions are protocol specific and typically use verbose identifiers.

Opponents to NATs also cite a lack of redundancy [2]. NAT is not in keeping with a stateless or transparently peering operation of network gateways. This increases costs, especially if redundant gateways are required.

Finally, with each addition to such a constrained network, the cost of migration increases. Unfortunately, economies of scale act against the introduction of IP6. As the value of IP4 addresses increase, lesser use ensures that IP6 remains more expensive. The additional benefits of IP6, such as addressing and multi-cast, are also a factor. However, if backbones remain as IP4 then IP6 must be tunnelled. This imposes performance limitations. Firstly, tunnelling increases packet sizes. Secondly, routing is not optimal. Thirdly, protocol stack tasks may be performed at application level, creating further inefficiency.

The Metaverse

Users request an intuitive and unified network interface. To address this problem, we have devised a simple schema for the publication of three dimensional worlds. Terminals present viewports within a shared environment. The schema itself is suitable for virtual hosting and could be offered in addition to virtual hosted web sites. Ideally, such a service could be offered as an extension to DNS:

1 DNS offers an extensible range of 65536 data-types. Very few have been utilised and a some have reached obsolescence.
2 DNS offers a convenient expiry mechanism which could used to provide simple dynamic content.
3 DNS scales and has a very large installed user-base. Existing infrastructure provides caching for millions of concurrent users.

DNS appears to be the logical choice for rapid deployment. This is especially true given the installed infrastructure. Unfortunately, DNS suffers from numerous drawbacks:

1 Delegation of DNS is poorly implemented. Resources required to resolve delegation are unbounded.
2 Delegation of DNS is insecure.
3 Payload is limited to 255 bytes before TCP is required.
4 International language support within DNS is exceedingly poor. This is extremely disappointing because DNS is eight bit clean. Limitation exists through incorrect implementation. Limitation also exists due to US interests [4].
5 Common implementations refuse to cache novel data-types. This is contrary to specification [5].
6 Migration to IP6 DNS is overlooked. Current proposals do not facilitate migration or adoption [5].
7 Replication of DNS configuration occurs via numerous methods. This hinders deployment of diverse implementations.

For these reasons, DNS was not adopted. This decision was not taken lightly. However, several benefits arise from alternatives:

1 Payload limits are alleviated. This greatly increases network efficiency.
2 Resource name limits are alleviated. This includes international language constraints.
3 Deployment of a separate service allows greater security.
4 A new service can be transport agnostic or specifically designed to facilitate migration.
5 A more fine-grained expiry mechanism facilitates real-time graphics.
6 Delegation of distributed resources can be implemented succinctly and securely.

DNS style protocols operate as a hierarchy of caches. This is compatible with a hierarchy of NATs offering caching facilities. However, several limitations remain within IP4:

1 Within IP4, delegation of servers occurs within an ambiguous address space.
2 Within IP4, peering of servers can only occur within the same tier. Unfortunately, this condition cannot be detected by interested parties. Therefore, this option must be discounted entirely.
3 Within IP4, packets have a 16 bit identifier to facilitate re-assembly of fragmented packets. Identifiers are chosen at source. Within a tiered NAT system, identifiers, in combination with source, destination and service, remain ambiguous. Collisions increase as bandwidth scales. UDP protocols containing their own request numbers are not exempt from this problem.

It is a real concern that dependance on IP4 will create a "produce and consume" model. Such a model will restrict resources to a narrow economic criteria. Alternative protocols will become increasingly economically prohibitive. Furthermore, unless we wish to create a trivial model of deployment, unique addressing of clients is required. Address workarounds vastly increase payload and may be protocol specific.

IP6 is a solution to all of the above problems.