April 2014

An Analysis of the SPDY Protocol and the SPDY Proxy

Executive Summary

Web-based commerce and content continues to grow, with mobile devices accounting for an increasing portion of network traffic. In a bid to enhance the speed of the web, browser and application developers are implementing new protocols to improve performance. At the same time, end-to-end encryption is increasing being used to improve user privacy and security.

Google's SPDY, described as "an experimental protocol for a faster web", is one of the new protocols being introduced to reduce web page load times. Deployment is increasing, but as of today many web servers still do not support SPDY. To accelerate adoption of the protocol, SPDY proxy services have been deployed to offer some of the benefits of SPDY even when downloading content from web servers that have not yet implemented the SPDY protocol. However, by bundling multiple requests and encrypting all web traffic for a user inside a single connection, the SPDY proxy creates an opaque tunnel, hiding the true source of the content and breaking network management, content distribution, and many services offered by the network operators. This would be equally true for any proxy service that bundled all traffic from a given user inside an opaque tunnel, irrespective of the protocol used.

The Problem

Web content is becoming more complex both in size and in the number of separate connections/requests. Web pages are dominated by images and scripts, both of which are easily compressible, but are not compressed by current implementations. Mobile devices over wireless networks are constrained in terms of bandwidth and latency. Unfortunately, protocols like HTTP (application level request/response) and TCP (reliable transport) were not designed to handle today's larger, composite web pages, and performance, especially page load times, suffers as a result. The trend toward encrypting all web traffic can further slow page load times, especially on mobile devices.

These factors provide motivation to speed up the web (performance optimization), especially to mobile devices. There is broad industry consensus that web page delivery optimization will be necessary to continue to provide a quality user experience. Building on early deployments of such optimizations, the Internet Engineering Task Force (IETF) has embraced the SPDY protocol as a starting point for standardization of HTTP/2 (expected Nov 2014). This white paper will analyze these developments, and assess the implications from two perspectives:

- **Protocol - SPDY:** by forcing end-to-end encryption, SPDY complicates network management, content distribution, and services offered by service providers. The use of end-to-end encryption inherently involves a tradeoff between privacy on the one hand, and access to network based services like malware detection on the other. Encryption also slows page load times on mobile devices, especially for "mash-up" sites such as news networks. Forcing end-to-end encryption does not allow the user to make their own decision about these tradeoffs.
- **Architecture - SPDY-based proxies:** encryption and opaque traffic bundling performed by SPDY-based proxies hides all metadata for the user's web traffic and breaks network management, content distribution and network services. When the traffic through opaque service optimization proxies is small, the effect is limited, but if it grows to dominate network traffic it will create fundamental problems for service providers and for consumers.

In addition, this paper will outline a path forward which could overcome the problems identified above.

The SPDY Protocol

SPDY has been developed and deployed to improve web page download times, but it will also have several impacts on end users and networks, including:

- Users will generally see faster web-page load time and improved web-site response for wired Internet access, but for mobile networks, the increased latency, and in particular, the variation in latency can dramatically reduce the benefit. In some cases, performance will actually be worse.
- SPDY will encrypt data streams, enhancing end user privacy, but limiting the ability of a network provider to perform network management functions such as performance data collection and video optimization. Encryption of all traffic also limits the ability of network providers to protect customers from web attacks. In high latency networks, encryption will create a slower web. For example, in the case of satellite, enabling encryption for all content often increases page load times from 5 seconds to 20 seconds.
- SPDY introduces new server-side controls, which claim to improve page load times by anticipating content that will be needed. In practice, these server-side controls could alter customers' utilization on usage-based data plans. Specifically, the Server Push function can push a resource to a client without the client first requesting such resource, and has the potential to confuse customers by impacting data usage in unexpected ways.

The SPDY-based Proxy

Before SPDY has been widely deployed on web servers, SPDY-based proxy services are being deployed to redirect all browser traffic through the proxy for "service optimization". Proxy refers to an intermediary between the client application and server application that processes requests from the client to server, and/or responses from the server to client. In the case of a SPDY proxy, the client application is a web browser and the server application is a web server. The SPDY proxy acts as an intermediary between the client browser and all Internet destinations, performing DNS requests, retrieving web content from the origin web servers on behalf of the client, and encapsulating all content inside an opaque tunnel to the client. The deployment of a proxy, illustrated in figure 1, dramatically increases the negative impact on content providers, networks and users:

- Redirecting all traffic through SPDY-based proxy services virtually eliminates the ability of the network provider to perform network management and content distribution which could degrade the network performance experienced by the end user. The loss of network management functionality results from both the creation of a single opaque tunnel, and from performing DNS queries at the proxy.
- SPDY-based proxies could significantly alter the traffic flow, creating new congestion at peering points, and potentially creating single points of failure. The SPDY proxy could "break" existing content delivery optimization mechanisms used by service providers, and as a result could further degrade service quality for consumers. It would also make it more difficult to isolate and resolve service problems.

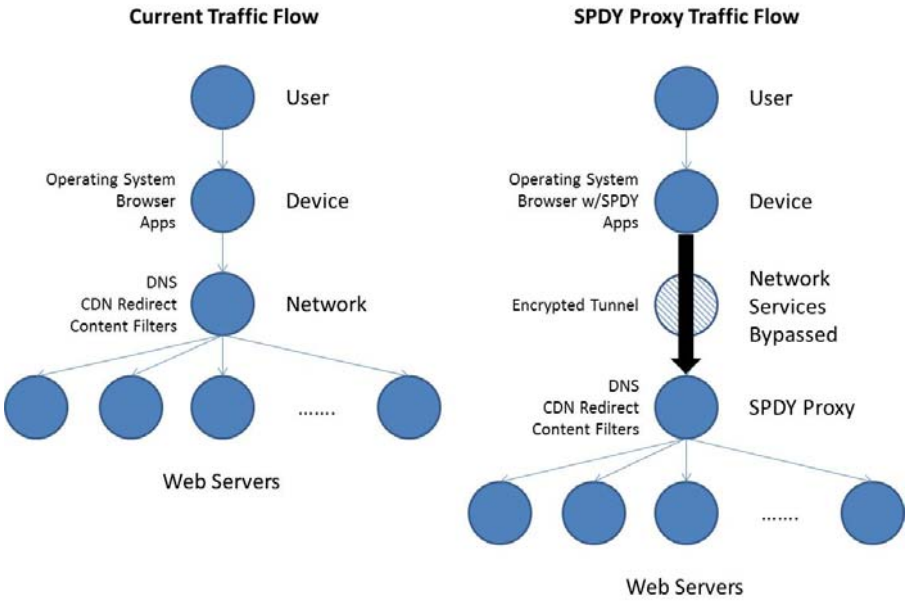

Figure 1: SPDY-based Proxy Traffic Flow

A Path Forward

Encryption, applied end-to-end from the user to the web server, is a valuable tool for protecting the user's privacy. However, encryption inherently involves a tradeoff between privacy and access to valuable network services. The tradeoff could be different for each user, and therefore the use of encryption should ideally be based on user choice. In addition, when a proxy is deployed in a way that redirects traffic into a single opaque tunnel across the network it severely hinders reasonable network management, content distribution and network services. This is an industry architecture issue that should be addressed in a collaborative forum.

Service providers could attempt to restore network management functionality by deploying open proxy-based solutions but this would introduce additional complexity and could degrade performance. This would also require cooperation between all parties, or the competing proxies could conflict, leading to unpredictable behavior.

Service optimization proxies, if they hide network utilization metadata, favor the organization that operates the proxy. Applications developers, content owners and service providers can each experience an advantage or disadvantage from service optimization solutions enabled in different parts of the ecosystem and operated at different layers. From the consumer's perspective, it would be preferable for the industry to cooperate to identify a solution that enhances security and performance for all parties while maintaining established relationships with the user. A coordinated industry approach could provide the consumer with a better understanding of the options available, clear choices, consistent behavior and superior performance.

This white paper proposes the formation of an Open Forum to conduct further analysis and to define solutions to these problems. To be effective, this effort should include all stakeholders in the broader ecosystem, including all entities with established trust relationships with the user. It would also include privacy advocacy groups, industry consortiums and regulatory bodies to reflect the current reality of Internet usage, based on trends, technologies and service deployments. This is outlined in more detail in the conclusions to this paper.

1 Introduction

A confluence of factors is contributing to an increased focus on techniques to speed up the web. First, web pages are becoming more complex and content-rich. For the top 100 web sites, the total transfer size of a web page now exceeds 1.5 MB, with over 90 connections/requests per page[1]. Further, the composition of web pages is increasingly dominated by images and scripts, both of which would be easy to compress, even though this is not generally done today (see Figure 2).

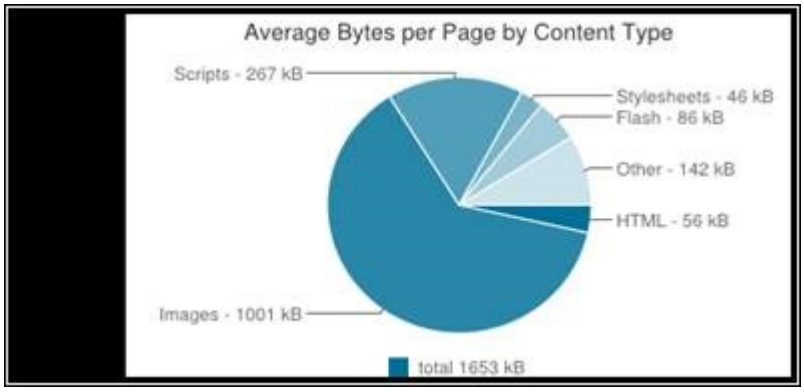

Figure 2: November 2013 web page composition for top 100 sites - Source: httparchive.org

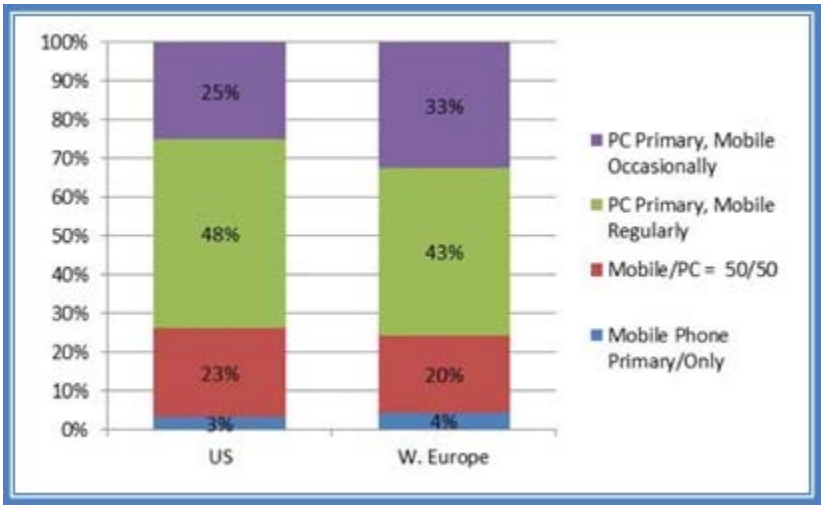

Figure 3: Consumer access to the Internet - PC vs Mobile (Strategic Analytics, Mobile Internet Behaviors: Browser and App Preferences - December 2012)

Second, web pages are increasingly being delivered to mobile devices over cellular (or mobile) networks which are constrained in terms of bandwidth and latency. With the proliferation of smart phones and tablets, customers expect the same, or even improved performance on these devices. Figure 2 shows that three quarters of US consumers use their mobile devices to access the Internet regularly and one quarter use mobile at least as often as they use a PC. And yet even the best-case bandwidth and latency performance of cellular networks is on average more constrained than most wireline networks; there are

[1] Source: httparchive.org

of course exceptions - e.g., from DSL to LTE - where the end user would see better performance on the wireless link. Third, protocols such as HTTP (application level request/response) and TCP (reliable transport) were not designed to handle today's larger, composite web pages. TCP's handshaking technique and implementations of HTTP attributes that limit each connection to a single request, begin to cause performance problems as mobile devices are accessing increasingly rich and complex content over limited bandwidth/higher latency networks.

When taken together, these factors explain the motivation to speed up the web (i.e., performance optimization), especially to mobile devices. Increases in network bandwidth and improvements to latency performance will help, but it will not be enough. Web page delivery optimization, in conjunction with continuing underlying network performance improvements, are both generally believed to be necessary to continue to deliver quality user experiences on the mobile web.

This white paper will examine two approaches to optimize web page delivery and consider the benefits, limitations, and issues with each approach.

a) **Client-server protocol:** With this approach, an end-to-end protocol is deployed to optimize web page delivery. There is a general recognition of the need to update HTTP/1.1 to better align with the reality of current web content; the IETF is developing HTTP/2 to address this need. To improve performance before a standard is available, a number of proprietary solutions, such as SPDY, have been implemented in portions of the network. Once HTTP/2 is available, it is expected that proprietary deployments will be withdrawn in favor of HTTP/2.
b) **Service optimization proxy:** In this case, all web traffic (or much of it) is redirected through an intermediate proxy that implements techniques to accelerate web page delivery. The appeal of this approach is that improvements to web page delivery time may be realized before the end-to-end protocol has been deployed on all web servers. However, forcing all traffic through proprietary proxy implementations can have unintended consequences when the proxy sends all traffic inside a single opaque tunnel. This is equally true if the proxy is based on SPDY, HTTP/2, or a completely different protocol.

It is important to distinguish between a protocol used end-to-end, and the same protocol used in an intermediate proxy. This is especially true for encryption. When it is used to protect the user's traffic end-to-end, the use of encryption can assure privacy. However, when encryption is used to a proxy the benefit is limited. It can protect the user's traffic on the access link – for example from a Wi-Fi hotspot, but it does not provide any assurance of encryption beyond the proxy. The potential benefit of encryption to an intermediate proxy is offset by the introduction of many problems, as discussed further in this white paper.

2 Protocols to Accelerate Web Content

Hypertext Transfer Protocol (HTTP) has been highly successful, supporting everything from web browsing and email, to innovative new and disruptive business models. However, the Internet has changed over the past decade, and HTTP/1.1 is showing its age. There is emerging consensus within the technical community that HTTP/1.1 needs to be updated to the current reality of network deployment and user expectations.

The IETF working group responsible for updating HTTP (httpbis) solicited proposals from the technical community on specific protocols that exhibit these general characteristics. Three candidate protocols were submitted[2], with SPDY being selected as the starting point for HTTP/2.

[2] The other protocols submitted were HTTP S+M (Speed+Mobility), and Network-Friendly Upgrade.

2.1 High Level Goals

The overall goal for HTTP/2 is simple: to reduce web page load time. It is generally accepted today that on wired and WiFi links, HTTP/2 will demonstrate 30-50% improvement in page load time when compared to HTTP/1.1. To minimize deployment complexity, HTTP/2 will continue using TCP as the underlying transport layer, and will be compatible with existing content; changes are confined to the browser and the web server.

2.2 Technical Goals

HTTP/2 is not a transport protocol in its own right, instead it is best viewed as a set of optimizations that allows multiple, concurrent, interleaved streams over a single TCP connection. Figure 4 depicts HTTP/2 and HTTP/1.1 protocols for comparison.

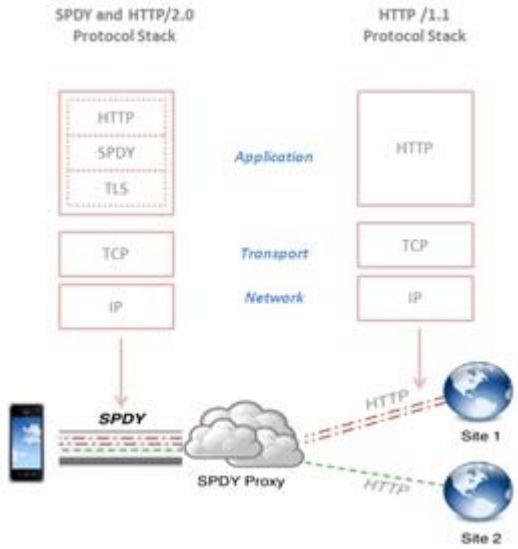

Figure 4: The SPDY/HTTP/2 and HTTP/1.1 protocols

The following attributes allow HTTP/2 to achieve reduced web page load time:

- *Multiplexing*. HTTP/2 allows multiple requests or responses to be sent concurrently on a stream.
- *Prioritization*. HTTP/2 allows prioritization of requests, letting the most important requests complete quickly, avoiding the head-of-line blocking problem.
- *Server push/hint*. HTTP/2 enables the server to pre-emptively send multiple resources to a client because the server can anticipate the need for these resources, reducing page load times as a result.
- *Compression*. A single web page includes multiple requests, and the redundant header fields in the requests use unnecessary bandwidth. Header compression provides an 80 - 90% reduction in these headers.
- *Encryption and authentication*. SPDY introduced “mandatory to use” security, i.e., all connections were encrypted. After its adoption as HTTP/2, the httpbis working group initially declined this use, but is now considering mechanisms that would dramatically increase the use of encryption in the network.

HTTP/2 will be deployed in both mobile and fixed networks. Mobile networks have significantly different characteristics than fixed networks, and this can lead to dramatically different behavior. For example, a recent analysis of SPDY by ATT Labs [Towards a SPDY'ier Mobile Network?[3]] identified fundamental interactions across the protocol layers, in particular TCP, which limit the performance of SPDY. The analysis concludes that while SPDY can enhance performance for fixed networks, there is no fundamental performance improvement for SPDY in mobile networks. The paper also notes that the interaction with TCP may be an underlying issue, which suggests that further analysis is required. A new protocol, known as QUIC is being introduced in an attempt to address these shortcomings, but this is beyond the scope of this white paper.

3 Intermediaries

A proxy, also known as a middlebox, is an intermediary that is inserted into the data path. Although the Internet architecture is nominally intended to be "end-to-end", with all functionality implemented in the end-points, there are many valid reasons to deploy proxies. These include:

- *Service provider*: Proxies are deployed to offer network services (e.g., parental controls), to provide reasonable network management (e.g., local caching of content) and security (e.g., malware detection).
- *Enterprise*: Proxies are deployed for traffic management (e.g., NAT, DNS) security (e.g., Firewall) and for secure networking (e.g., VPN).
- *Application providers*: Proxies are deployed to efficiently deliver applications (e.g., load balancing) and security (e.g., Firewall).
- *Split browsers*: SPDY proxies deployed by Google/Amazon as an extension of the browser are also proxies.

In general, each of these proxies is operated by a single entity, and is "closed" (i.e., no other entity has visibility to the functions performed by the proxy). In addition, the functionality provided by the proxy is not specified, which can introduce incompatibilities with browsers and applications. Historically, each proxy has operated within a well-defined domain, so there has been little interaction. However, increasing browser options, a proliferation of applications, and additional proxies all increase the risk of incompatibilities. The current trend raises the possibility of "competing proxies" that can interact and interfere with each other. In response, we are proposing the concept of an "open service optimization proxy" which could be managed in a way that could satisfy the requirements of all entities, while still maintaining established business relationships with the user. The characteristics of an open proxy would be fully defined in an open forum. A reference implementation could be developed, and made available for interoperability testing to ensure compatibility with browsers and applications. The broad attributes of an open proxy are discussed in this white paper, but fully defining the requirements would involve additional collaborative study under the direction of an Open Forum.

3.1 SPDY-based Proxy

While many of the largest Internet destinations have already adopted SPDY (e.g., Facebook, Gmail, Wordpress, and Twitter), the benefits perceived by content owners (encryption for consumer privacy, data compression and faster page load times), as well as the potential to capture an larger set of data analytics, have prompted some organizations to implement SPDY-based proxy services.

Like any other proxy, SPDY proxies act as an additional middlebox in the data path. The SPDY proxy creates a tunnel between the browser and the proxy, and carries the HTTP sessions within the SPDY

[3] < http://conferences.sigcomm.org/co-next/2013/program/p303.pdf >

tunnel, extending support for SPDY over the access link for all destinations when requested by a supported browser. In this instance, the SPDY proxy could open multiple HTTP requests from the proxy to multiple destinations, get the responses, and aggregate the responses back to the browser over a single TCP connection. This can be useful for websites that aggregate content from multiple destinations, allowing a SPDY proxy to multiplex HTTP requests to multiple destinations over a single TCP connection, eliminating multiple TCP handshake overhead and TCP slow-start computations between the client and server over the access network. In addition to the inherent benefits of SPDY, the proxy functionality can potentially reduce the amount of data consumed by the mobile subscriber against their tiered data plans.

The organizations providing SPDY-based proxy functionality today include Google (for the Chrome browser) and Amazon (for the Silk browser). In both instances, the SPDY proxy functionality is hosted in the respective company's datacenters, although Google may also offer the SPDY proxy functionality within their Google Global Cache platform. Both implementations of SPDY proxy are technically opt-in services for the consumer, but even advanced users may not fully understand the implications of enabling the SPDY proxy.

The following figure illustrates the Google SPDY proxy[4] as an example.

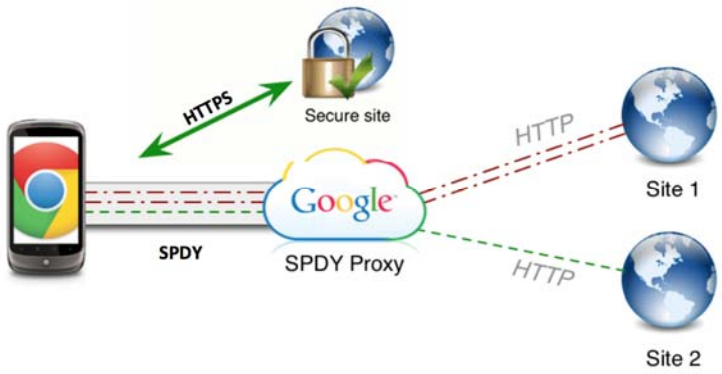

Figure 5: Google SPDY Proxy

The model shown in Figure 5 provides a number of potential opportunities to the organization that operates it:

- Provides complete visibility into all web traffic traversing the SPDY proxy, not just their own traffic.
- Eliminates the need for client devices to resolve HTTP destinations via DNS and removes DNS signalling from the mobile airlink, since the SPDY proxy performs the DNS lookups. However, this breaks the effectiveness of existing DNS-based approaches to value added services (VAS) on the client.
- Dramatically changes traffic patterns, increasing the inbound/outbound traffic from the proxy, thereby potentially influencing peering relationship negotiations.
- Creates the opportunity to provide web acceleration by implementing CDN caching functions within the SPDY proxy. However, this makes it difficult or impossible for service providers to effectively implement CDN caching, typically much closer to the user, for traffic that uses the SPDY proxy. In the case of Google, the SPDY proxy functionality is already coupled with the PageSpeed[5] service for web optimization (Image transcoding and content compression).

[4] https://developers.google.com/chrome/mobile/docs/data-compression

[5] < https://developers.google.com/speed/pagespeed/ >

- Allows for a co-processing model of HTTP destinations that can improve performance and battery life on mobile devices. For instance, in the case of Silk, Amazon can determine which browser subsystems (networking, HTML, page rendering, etc.) run locally versus in the AWS cloud[6].

These proxies also have implications for service providers from both a technical and business perspective, including:

- Changes peering relationship negotiations with SPDY proxy owners, due to the increased inbound (to end users) and outbound traffic (from Communications Service Providers [CSP]).
- Reduced visibility into network traffic, including lost visibility into individual Internet flows and lost visibility into DNS requests. This will reduce the ability of service providers to implement reasonable network management.
- Inefficient content routing, since the majority of Content Delivery Networks (CDNs) rely on a combination of DNS Requests and Source IP Address to return IP ranges. In the case of the DNS late-binding mechanism used by SPDY proxies, the Source IP address issuing the DNS request would refer to the SPDY proxy (and the SPDY proxy's ISP), which might not be illustrative of the closest cache from which to serve content. This may directly impact the latency perceived by the user.
- Modification of traffic flows, potentially leading to different capacity planning models on a per-node and network-wide basis. SPDY changes flow sizes and durations, leading to fewer, higher-bandwidth flows that impact network and infrastructure design.
- Changes the VAS paradigm by (a) bypassing carrier VAS services, (b) coupling caching, video optimization, image compression and web acceleration with the SPDY proxy itself, and (c) enabling co-processing model for HTTP that eliminates well-known application protocols from traversing the service provider network.

3.2 Service Provider Proxies

One potential scenario in which the Service Providers may regain some level of visibility is through the implementation of a SP-owned service optimization proxy. These proxies could be SPDY-based to accelerate SPDY deployments and support the OTT and content ecosystems in a way that is friendly to many of the SP business and technical practices that exist today. Such a SPDY proxy would provide functions similar to existing SPDY-based proxies, but in a content-neutral way, without violating any end-to-end security or consumer privacy paradigms required by content and application owners.

Deploying a consistent SPDY proxy capability across multiple locations, application providers, and Service Providers alleviates many of the concerns raised in this paper regarding third party SPDY proxies. These concerns are resolved in the following ways:

- The asymmetric peering relationships that exist today remain the same, since the SPDY proxy would exist within the domain of the Communication Service Providers (CSP).
- Network traffic, at least at the per-flow level, would be visible to the CSP, as the SPDY proxy owner, allowing traditional network management, security, optimization, and monetization models based on DNS and 5-tuple information (i.e., source IP address, destination IP address, source port number, destination port number, and protocol) to continue. The content of the flows, however, would remain encrypted, and the end-to-end security paradigm between the client and content provider remains.
- DNS requests would originate from the CSP SPDY proxy, allowing CDNs to operate as they do today. In fact, Service Provider SPDY proxies would reduce the number of paths through the CSP infrastructure in a way that would allow CDNs to more-consistently distribute content to the right cache.

[6] < http://www.youtube.com/watch?feature=player_embedded&v=_u7F_56WhHk >

- While minor modifications to network systems and design between the client and CSP SPDY proxy may be required, there would be no impact on systems and design between the CSP SPDY proxy and the Internet.
- Many VAS functions provided by the CSP, especially those that only require visibility into DNS requests and 5-tuple information, would retain their value, and remain available to users interested in these services.

The implementation of a SP-owned SPDY proxy would require modifications to devices, browsers and/or applications to allow the user to discover available proxies, select the desired proxy, and steer traffic to the proxy. Today's client implementations of SPDY proxy functions do not allow the explicit discovery, selection and configuration of a SPDY proxy. In the absence of a discovery mechanism, the SP or the device owner would need the ability to assign the SPDY proxy address, either by DNS Name or IP address.

More importantly, to be effective, the SP-owned SPDY proxy would require the elimination of any other SPDY proxies from the data path, to eliminate the inefficiencies in layering tunnelling headers and restore the ability to view header information for individual destination flows. The SP-owned SPDY proxy provides a mechanism to realize the efficiencies of SPDY over access networks, especially mobile access networks, while re-enabling functions such as reverse DNS resolution and 5-tuple based VAS services from the SP.

Lastly, careful attention to the impacts of mobility, especially involving multiple mobile ISPs (LTE roaming relationships, WiFi, etc.), on the CSP service optimization proxy model is required. Static configuration of a proxy within the originating SP domain creates inefficient routing for client devices whose point of attachment resides outside that SP's domain. This holds true whether the mobility is seamless (handoffs between access networks) or nomadic. Requiring all traffic to return to the client's home SP proxy increases latency when content and applications can be more-effectively delivered via a local system. It is therefore important that access to a service optimization proxy be configurable, and under the control of the communications service provider to optimize for the network location of the client device.

4 Network Services

Service providers offer a number of network management and value added services today. In general, these network services are provided by "middleboxes".The tasks performed by these middleboxes can be divided into four broad categories: packet inspectors, content modifiers, protocol proxies, and Value-Added Services (VAS). Examples of such platforms and associated use-cases are identified in the following table.

Table 1: Optimization Services

Service	Examples	Use Case
Packet Inspectors	Deep Packet Inspection IPFix Network Probes Network Firewalls Web Application Firewalls Network Security (IPS, DDoS Detectors)	Traffic/Activity Monitoring Data Analytics L2-L7 Load-balancing Behavior Analysis Anomaly Detection DoS/DDoS Protection Malware Detection
Content Modifiers	Content Optimizers Protocol Optimizers	Transrate/Compress Video Image Compression

Service	Examples	Use Case
	Compressors	Optimize TCP Slow-start
Protocol Proxies	Network Address Translators DNS Cache SIP Proxy Session Border Controller HTTP Proxy WebRTC Gateway TCP Proxy	Modifying IP address information Communications Control Content Caching ABR Index Modification Performance Enhancing Proxy L7 Application Functions
Value-Added Services	Ad Insertion Engine Header Insertion	Advanced Advertising In-stream header enrichment URL Filtering Parental Control

Proxies, and the network services they provide, can also be viewed from the perspective of the end user, and the problems they are trying to solve. For a discussion of proxies from the user's perspective, see: https://github.com/http2/http2-spec/wiki/Proxy-User-Stories

Middleboxes function by inspecting or modifying various headers as traffic passes between the requester (client) and responder (server). This depends on the ability of the middlebox to see the appropriate headers.

The impact that SPDY has on services provided by middleboxes depends on the service and how SPDY is deployed in the network. Specifically:

- In some cases, SPDY actually reduces the need for specific middlebox functionality by providing inherent support for capabilities such as TCP Slow-Start and window optimization. However, many other middlebox functions are still required.
- Many of the transparent middleboxes that operate mainly at Layer 4, such as NAT, firewalls, and load-balances, are largely unaffected by the use of SPDY when it is used end-to-end between the browser and the server. In other cases, the carriers' ability to provide VAS to their customer base for Internet traffic is reduced by SPDY, but is not eliminated.
- When a third-party SPDY-based proxy is introduced into the network, it bundles all traffic in a single opaque tunnel, which precludes virtually all value added services and reasonable network management functions by the carrier. This could lead to increased network congestion and difficulty isolating network failures.

Deployment of third-party SPDY-based proxies seriously impacts the ability of service providers to offer network services and to implement reasonable network management, including transparent video caching. However, if the SPDY proxy is deployed by the service provider, as discussed in the previous section, then network management, and many network services, will continue to function as they do today.

5 Consumer Implications

This section will focus on the implications of SPDY-based proxies for consumers. The consumer impact falls into the following broad categories which will each be considered here:

- Security
- Privacy
- Network services
- Performance
- Choice

Security: As outlined in the previous sections, SPDY natively incorporates transport level security in the form of SSL/TLS. The technical rationale for this encryption is to prevent middle functions from disrupting one or more http(s) sessions that are multiplexed by SPDY. The added benefit is the encryption of traffic between the SPDY-enabled browser and server. Given rising user concerns about surveillance, this transport-level encryption provides a mechanism to secure unprotected http traffic. The process is similar to that of https except that SPDY will encrypt http traffic as well. In practice, however, with the use of the SPDY proxy, there is no guarantee that the http content will continue to be encrypted when it is proxied by the SPDY server to its final destination. The http content can also be seen by the proxy. Encryption to the SPDY-based proxy does not guarantee end-to-end security. Fundamentally, https secures a specific application while a SPDY-based proxy is securing a path.

Privacy: The use of SPDY-based proxies may have additional privacy implications for consumers. As outlined previously in the security section, the encryption between the user and proxy provides some protection against surveillance on that path, but it does not provide protection after the traffic leaves the proxy. The browser settings also do not make it clear that the user is choosing to redirect their requests through the proxy. With Android, the setting is titled “Reduce data usage” under the “Bandwidth Management” settings. With Amazon Silk, the setting is “Accelerate Page Loading”. In each case, the feature is bound by the current privacy policies of the respective company, and additional consumer information may be collected based on the redirected traffic. Per the *Federal Trade Commission Report on Protecting Consumer Privacy in an Era of Rapid Change* published in March 2012, “For practices requiring choice, companies should offer the choice at a time and in a context in which the consumer is making a decision about his or her data. Companies should obtain affirmative express consent before using consumer data in a materially different manner than claimed when the data was collected.”

Network Services: In general, the addition of encryption is hard to argue against, but it is important to recognize that encryption is also only one aspect of a multi-factor security plan, and that security is only one aspect of service delivery. A SPDY-based proxy will bypass intermediate network services such as firewalls, content filters, malware-detection, and parental controls. This can allow malicious or other unwanted content to be delivered, while having little or no benefit for existing SSL and DRM-protected content.

Performance: The use of a SPDY-based proxy as described in Section 3 can cause additional complexity because of potential traffic redirection that occurs without user, network provider, or web provider awareness. This proxy also introduces an additional point of failure that is not visible to the user or website provider. Current implementations of SPDY selectively redirect traffic from the browser to an external proxy that terminates the encrypted path. The traffic then proceeds in native (non-SPDY) form to its final destination. This implementation provides some benefit by securing the last mile, but it does this at the expense of operational complexity and bypassing of intermediate network management functions. Existing mechanisms to improve network performance (e.g., caching) will not work. In some cases, traffic that is routed to the SPDY-based proxy will need to backtrack to reach the target web server, increasing latency. As a result, while SPDY could improve webpage load times, the use of SPDY-based proxies could actually reduce performance and make it more difficult for the user to resolve problems.

Choice: The user makes multiple choices in the process of accessing services including device, application and the network provider. In some cases, these choices, such as the device, operating

system, and browser, are bundled together. By routing all web traffic through a SPDY-based proxy, existing relationships between the user and other service providers are disrupted. Network-based services such as parental controls, malware detection, and network-based firewalls are examples of existing services that are impacted. As a result, SPDY-based proxies effectively limit consumer choice.

6 Conclusions / Recommendations

The protocols that power the Internet were designed for much simpler content and applications than we see today. These original protocols are still in use but they are layered vertically and horizontally to allow many companies to participate in a single service request. This open integration of web, network, content, and other providers is now standard practice in the internet.

In a bid to enhance the speed and security of the web to deal with the current reality, browser and application developers are implementing new protocols, including transport-layer encryption. Browsers with the new embedded protocols are changing the effective Internet architecture, rerouting network traffic, and bypassing existing intermediate services thereby creating new challenges for other parts of the Internet value chain.

These changes in architecture also come with changes in user experience. In particular, the deployment of proxy-based solutions bypasses many established services such as parental controls, network-based malware detection, and network management functions. SPDY proxies are implemented in a way that is ambiguous to all but expert users, and divert traffic away from the intended destination. If users do experience issues, it is unlikely that they will contact the browser manufacturer for assistance. The first call will more likely be either directed to the network provider or website provider who has no visibility to the redirected traffic. The primary impact will be that the provider will not have the information necessary to resolve the user's problem.

Once the deployment of SPDY proxies reaches a significant level, it will make it effectively impossible for service providers to see any information about the type of traffic on their networks. This will affect the ability of service providers to implement transparent video caching and other reasonable network management practices, which will begin to impact the performance of networks. This will have significant implications for the level of service received by consumers. It is possible this could begin to happen in the near future.

Competing proxy-based solutions could be deployed by service providers and other parties, but this would introduce additional complexity and could further degrade performance. In some cases, the competing solutions could even conflict, leading to unpredictable results. From the consumer's perspective, it would be far better for the industry to cooperate to find a solution that enhances security and performance in a way that ensures complementary, rather than competing solutions. A collaborative and coordinated industry approach could provide the consumer with clearer choices and a better understanding of the options available.

Service optimization proxies, such as the SPDY proxy, give power to the entity operating the proxy. Applications developers, content owners or service providers can each experience an advantage or disadvantage from service optimization solutions enabled in different parts of the ecosystem and operated at different layers. Consumers want solutions that are secure, fast, and offer a wide choice of services and applications. The way to provide this is through a coordinated approach to service optimization proxies.

Standards organizations are actively working on optimized protocols for specific needs. These protocols only optimize one aspect of the ecosystem, and as a result there is potential for conflicting behavior when implemented in actual networks. Protocol work needs to be balanced with a broader view of how these standards-based solutions will interact in the Internet ecosystem and especially how they will affect the consumer. A broad engagement that includes all segments of the Internet business ecosystem is needed to consider this issue and look for collaborative solutions that can be embraced by all stakeholders and standards organizations.

To be effective, this effort should include the following elements:

- Collaboration with the broader ecosystem stakeholders to identify the requirements for solutions that do not conflict with each other and that do not infringe on the established trust relationships with the user. Today, solutions are being developed and implemented independently, without explicit consideration of the broader ecosystem impacts.
- Collaboration with privacy advocacy groups to promote the use of secure communications and educate the user community on the various aspects of security including trust, user consent, and transparency.
- Collaboration with regulatory bodies in realignment with the current reality of Internet usage, based on trends, technologies and service deployments.

To facilitate this collaboration, ATIS is sponsoring the Open Web Alliance, an industry-wide forum to develop requirements for an open service optimization proxy that would accommodate both web and network services while supporting the goals of encryption and privacy. The Open Web Alliance invites a broad range of participation including web providers, network operators, and device manufacturers. Global participation in this activity is desirable. ATIS membership is not a requirement.

The requirements developed by the Open Web Alliance will be contributed to standards organizations such as the IETF for the purpose of establishing an open architectural framework and associated protocols to support a secure, distributed service model. The requisite advocacy of the contributions would be done to facilitate deliberation and secure adoption.

This paper was prepared by ATIS. For more information, contact Jim McEachern, ATIS Senior Technology Consultant, at jmceachern@atis.org.