# Apple FaceTime on Multimedia-Grade Aruba WLAN

## Table of Contents

## Introduction

As the corporate world increasingly draws on innovations from the consumer market, visionaries ranging from Zeus Kerravala[1] of Yankee Group to Lucy Kellaway[2] of the Financial Times are predicting that 2011 will be the year of corporate video. Not unexpectedly, unified communications and collaboration vendors, including Cisco, Avaya and Polycom, are also bullish on the potential for video to improve communications and productivity: All have announced products to support peer-to-peer video calling from desktop and mobile devices.

But the vendor with the largest footprint of deployed video-capable devices found in the enterprise is Apple, with its FaceTime application that runs on nearly all current Apple devices, including iPhone 4, iPod Touch (the new iPad will likely be FaceTime capable in early 2011), and now as an application on MacBook platforms.

Although Apple has not to date specifically targeted the enterprise market, its devices are found in growing numbers within corporate networks. Some IT groups are developing applications for iPhone, and formally supporting them, while many others accept that a significant percentage of employees will bring their iPhones to work and conduct business on them.

Aruba Networks has heard from many of our customers that wish to provide a secure, supported environment for Apple devices, and as a result we have developed several features specifically for mobile Apple devices connected over Wi-Fi. A companion paper deals with a new ArubaOS feature, Aruba Device Fingerprinting, where, on initial authentication to the wireless LAN (WLAN), devices are automatically categorized and assigned to appropriate roles.

This paper deals specifically with considerations and features that improve FaceTime performance on an Aruba WLAN. FaceTime is a Wi-Fi specific service: While a FaceTime call can be initiated from the cellular network, it is set up and carried on the WLAN. While it would be technically possible to run FaceTime over cellular data channels, operators' concerns about the capacity of their networks have driven this restriction: If too many users ran FaceTime over cellular data, cellular networks would be quickly overloaded.

FaceTime is the first of a number of peer-to-peer video calling services that will be rolled out in the near future. Video, as a high-bandwidth multimedia service, is one of the more challenging types of traffic to support on a WLAN, but current enterprise-grade WLANs are well able to deliver superior performance for FaceTime and future video-calling applications.

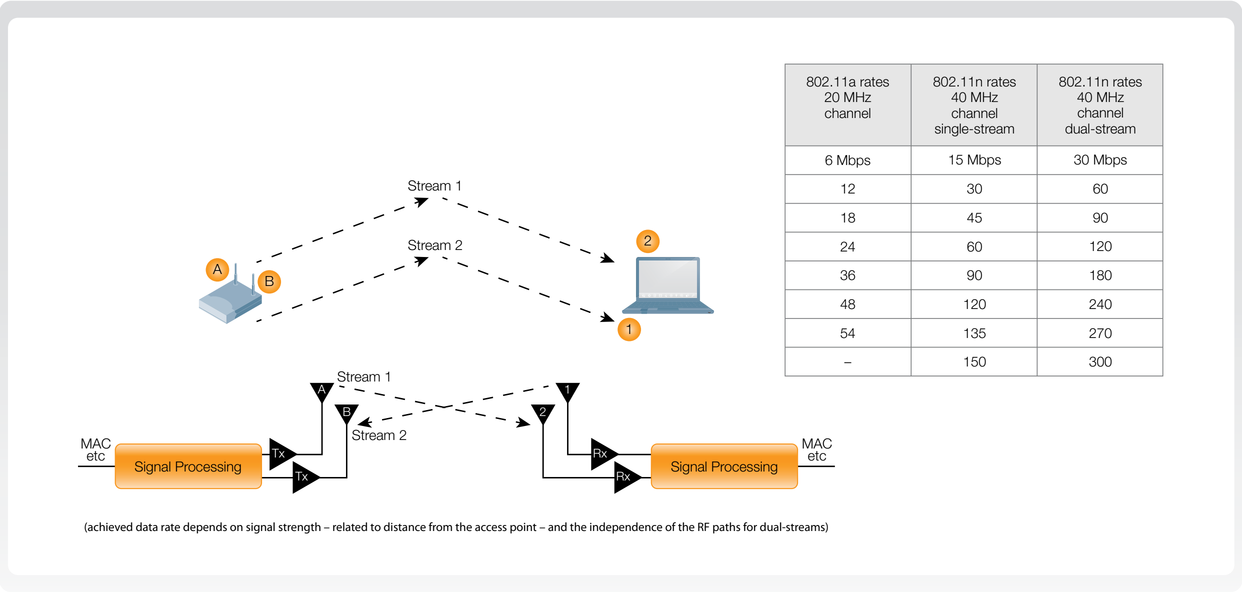
## Running FaceTime on an Aruba WLAN

### FaceTime protocols and procedures

The first step to using FaceTime is to connect the Apple device to the WLAN. FaceTime is very flexible: Any authentication method used to connect to the WLAN and the Internet will work.

As soon as it has an Internet connection, the Apple device registers with Apple's FaceTime servers. This registration establishes a binding between the device's address, whether email or phone number, and its IP address. Now anyone can call the device; for outgoing calls it can reach the Apple directory to find any other FaceTime device. The registration procedure is protected by mutual authentication based on certificates on the Apple server and the Apple device: It uses the same ports and protocols as Apple's iChat (Jabber) application.

Next, the FaceTime application automatically negotiates the required network address translation (NAT) and firewall traversal using Apple's servers. This involves STUN and TURN, techniques that allow the Apple servers to identify the internal/external address combination that will be able to reach the device, and to open up a firewall port from inside the enterprise network. NAT and firewall traversal are complicated, but well understood by now, and as far as we can see, Apple uses standard ports and protocols for this function.



| 802.11a rates 20 MHz channel | 802.11n rates 40 MHz channel single-stream | 802.11n rates 40 MHz channel dual-stream |
|---|---|---|
| 6 Mbps | 15 Mbps | 30 Mbps |
| 12 | 30 | 60 |
| 18 | 45 | 90 |
| 24 | 60 | 120 |
| 36 | 90 | 180 |
| 48 | 120 | 240 |
| 54 | 135 | 270 |
| – | 150 | 300 |

(achieved data rate depends on signal strength – related to distance from the access point – and the independence of the RF paths for dual-streams)

***Figure 1***

FaceTime uses two addressing modes[3]. The iPhone devices are reached using their phone number. Other devices must set up an email account and register the account with Apple's registration server (there is a 'settings' option for this, Apple calls it 'obtaining an Apple ID'). Any email address can be used, but it must be registered before FaceTime will make or receive calls. Apple's servers maintain a directory binding the user's phone number or email address to the IP address and routing information derived from STUN.

After the initial registration and STUN exchanges, the device waits for an incoming or outgoing call. When a user initiates a FaceTime call from an Apple device, the application first sends a query to the Apple registration server to find the IP address of the target. This is an encrypted exchange.

With this initial IP address information, the Apple device enters into a STUN, TURN and ICE exchange with the called device to establish a set of IP addresses and ports for the FaceTime session. This is a complicated process, but not Apple-specific, nor proprietary.

At the end of the STUN, TURN and ICE exchange, an INVITE message is sent by the calling device. This is a standard Session Initiation Protocol (SIP) call setup request including the name of the individual to be called, and a Session Description Protocol message (SDP) defining call parameters and bandwidth requirements. Subsequently, the call is set up using SIP, including 100 TRYING, 180 RINGING and 200 OK messages.

Immediately following call setup, a series of SIP MESSAGE packets are seen, along with exchanges with the Apple server. We believe these are used to mutually-authenticate the calling and called devices.

Once the SIP exchange has completed, the media stream is set up. FaceTime uses one UDP port for signaling and media: If SIP uses port 16402, the media will use the same port. Within that port, two separate streams are established: Video uses the H.264 codec, while the audio uses AAC. The sessions can be identified by their payload type and sequence numbers.

The call can be terminated by either end, using a SIP BYE message.

### Setting firewall rules for NAT and firewall traversal

Apple uses standard STUN, TURN and ICE ports and procedures to discover NAT and traverse the firewall: These procedures are described in RFCs 5389, 5766 and 5245 among others. The network must allow these ports to be opened from within the firewall, as is the case for most networks today. Perhaps just as interesting, this information could be used to prevent FaceTime activity – a block on connections to the Apple FaceTime servers' IP addresses should disable the service.

The table below shows the protocols and ports used by FaceTime. It is taken from Apple's Knowledge Base4 and verified by observations in Aruba. It is important to use Apple's data because it shows the complete range of ports that may be used. For instance, even though in most cases the FaceTime media streams use UDP port 16402, the table shows that when this is unavailable, ports down to 16393 and another range from 16384 to 16387 can be used.

| Port | Packet type | Used by protocol | Application (in the Apple domain) |
|---|---|---|---|
| 53 | TCP/UDP | Domain Name System (DNS) | MacDNS, FaceTime |
| 443 | TCP | Secure Sockets Layer (SSL, or "HTTPS") | Secured websites, iTunes Store, FaceTime, Game Center, MobileMe (authentication and MobileMe Sync) |
| 3478-3497 | UDP | – | FaceTime, Game Center |
| 5223 | TCP | XMPP over SSL, Apple Push Notification Service | MobileMe (Automatic sync notifications) (see note 9), APNs, FaceTime, Game Center |
| 16384-16387 | UDP | Real-Time Transport Protocol (RTP), Real-Time Control Protocol (RTCP) | FaceTime, Game Center |
| 16393-16402 | UDP | Real-Time Transport Protocol (RTP), Real-Time Contro Protocol (RTCP) | FaceTime, Game Center |

Aruba's observations indicate that corporate firewalls should allow TCP/UDP ports 53, 443, 3478-3497 (the usage of these is unclear) and 5223, as above: These ports are used for communication with Apple's FaceTime servers. The UDP ports 16384-16387 and 16393-16402 are for the signaling and media streams used by FaceTime: These need not be enabled for intra-corporate communication, but should be allowed for general FaceTime operation.

### Ensuring sufficient bandwidth for FaceTime

Bandwidth in modern enterprise LANs and WLANs is now quite plentiful, particularly with the uptake of 802.11n Wi-Fi access points (APs) and devices (including the iPhone 4 and iPod touch fourth generation). But the prudent network engineer will verify that new services, especially video services will not be bandwidth-limited on the network, nor starve other, possibly more important applications of bandwidth. This requires some knowledge of the required data rate.

As noted above, FaceTime separates the voice and data media streams for transmission: There are four media streams in each call. Voice streams use the AAC codec, apparently in constant or nearconstant bit rate mode, with a measured 68 kbps in each direction for a total of about 136 kbps over the air for the audio stream. Frames are around 180 bytes long over the air (the UDP payload is about 120 bytes and actual audio samples carried take up 88-92 bytes per frame).

The video stream uses the H.264 codec, also known at MPEG-4 Part 10 or AVC. H.264 has many options and rates, and it is not immediately clear which FaceTime uses, but the data rate is highly variable, ranging from 200 to 1000 kbps. Frames can contain payloads up to about 1200 bytes.

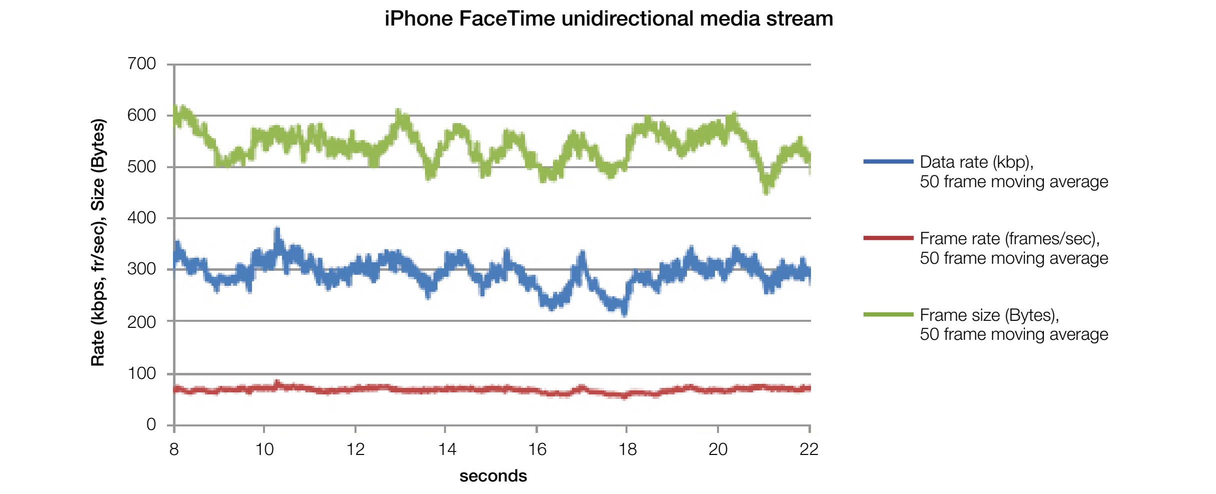


*Figure 2*

With variable rate codecs such as H.264, the more the picture changes from frame to frame, the higher the encoded data rate. The chart below gives a view of FaceTime data rates through a mobile call.

In planning for widespread FaceTime deployment, a representative bandwidth requirement would be around 350 kbps in each direction, for a total of 700 kbps per call, including both video and audio streams. Since nearly all FaceTime-capable Apple devices use 802.11n (only a few older MacBooks in the installed base will still be 802.11a/g), we consider representative figures for 802.11n networks, and for 802.11a/g infrastructure.

An 802.11n AP, with single-stream operation in 2.4GHz frequency band to Apple devices at a very conservative average modulation rate of 28.9 Mbps (the 802.11 maximum link rate, when close to the AP, is 65 Mbps) will be able to support approximately 16 simultaneous FaceTime calls.

The figures above take into account all Wi-Fi effects such as contention, retries, actual modulation rates (Mbps figures are taken from actual walkaround tests) and management traffic. But they assume that FaceTime is the only traffic on the AP: If other critical services such as voice-over-Wi-Fi are supported, lower number of Facetime calls per AP would be supported.

It is also helpful to make use of the available Wi-Fi spectrum with band-steering, an Aruba feature. The iPhone and iPod devices are 2.4 GHz-only, so they must use that part of the spectrum. However, MacBook and iPad (though the current generation does not support FaceTime) devices can use 2.4 GHz or 5 GHz, and should generally be steered towards 5 GHz, as there is less interference and nearly always more bandwidth available in this band.

## Enabling good QoS for FaceTime – Application Fingerprinting

As a multimedia service that is affected by delay, jitter and packet loss, FaceTime requires good end-to-end quality of service. Calls can take different paths, depending on the destination, but we can identify three regions calls may traverse: The WLAN, LAN and Internet. Each has its own quality-of-service (QoS) mechanisms, and the call should maintain QoS mapping at transitions for good end-to-end call quality.

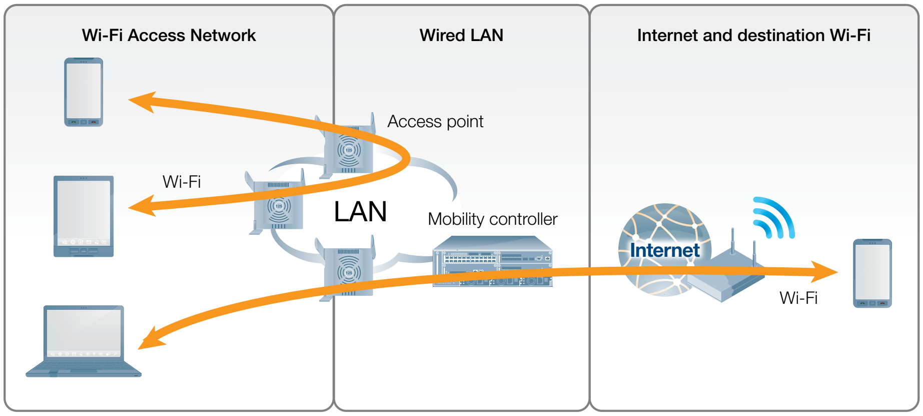


*Figure 3*

Over the Internet, the QoS mechanism should be IP TOS, an L3 tag. Over the LAN, QoS is applied using 802.1p tags, and over the WLAN the mechanism is Wi-Fi Multimedia (WMM). In our ideal model, the originator sets the appropriate QoS priority tags on packet headers, and tags are translated appropriately at network boundaries.

However, many devices and networks are not correctly configured for QoS. In the case of the FaceTime application we can identify a number of issues:

- Apple devices do not set the appropriate WMM priority when transmitting Wi-Fi frames on the WLAN.
- Some corporate LANs are not configured to correctly honor 802.1p priority, or sometimes 'lose' tags.
- The Internet does not generally honor or carry IP TOS tags.

This can create issues for FaceTime. In fact, since Apple devices do not set the appropriate priority when transmitting frames, FaceTime traffic is normally carried 'best effort' along with email, web browsing and general data traffic. For some network engineers, this may be a relief. At least FaceTime users will not starve others of bandwidth. But where FaceTime is seen as a useful business tool, we can and should do better.

Aruba has developed tools that identify FaceTime traffic and assign it appropriate QoS priority, for much but not its entire route. The feature, known as 'application fingerprinting,' uses Aruba's integrated stateful firewall to inspect and, where necessary, modify traffic. In this case the modification is slight – just the priority tag in the header.

Application fingerprinting is a two-stage process. First the WLAN identifies candidate devices to monitor for multimedia traffic. Then, when these devices transmit multimedia frames, it adjusts the QoS in both directions of the call to the appropriate priority.

The initial device identification is quite flexible: For Apple devices, we would normally use the ArubaOS Device Fingerprinting for Aruba Mobility Controllers that recognizes certain device characteristics at the association phase: Device Fingerprinting can identify iPod, iPad, iPhone and various MacBook devices, as well as other vendors' mobile devices. Alternatively, it is possible to recognize events such as registration to Apple's FaceTime server through its unique protocol.

Once devices have been identified as multimedia capable, the mobility controller continually examines their traffic to see if it decodes as RTP, with the known characteristics of a multimedia session. RTP streams are not easily identified: There is no single field in the packet header identifying them as RTP. Aruba uses its integral stateful firewall to continuously monitor multimedia devices, scrutinizing headers for several signatures of RTP streams, and a specific template that identifies FaceTime multimedia sessions.

With the sessions identified, Aruba resets the priority level. The two points of control in an Aruba network are the AP and the mobility controller. We can trace the path of both upstream and downstream traffic for QoS.

In the upstream direction, as soon as frames from the air are received by the AP, they are re-tagged to the correct priority. This tagging is set on the outside of the tunnel from the AP to the mobility controller, so QoS will be maintained across the LAN to the data center.

Likewise, northbound from the mobility controller's core network interface, the correct Layer 2 and Layer 3 priority is applied. In the case of Remote Access Points (RAPs) on bandwidth-limited backhaul, Aruba reserves uplink bandwidth for multimedia traffic, so FaceTime will be prioritized on this segment of the connection.

Downstream, traffic typically enters the mobility controller from the Internet or corporate WAN, and will be immediately re-tagged if necessary. The tagging is maintained over the LAN from the data center to the AP, and over the air priority is enforced using WMM.

This feature cannot correct priority on every link in the FaceTime path. Notably, over-the-air transmissions emanating from the Apple device towards the AP cannot be re-set until they are received, so they must contend with other traffic. Also, we assume the Internet will not maintain priority for the far-end segments of outbound calls. But most of the call is now covered, end to end.

In addition to retagging for correct QoS, Application Fingerprinting triggers a number of multimedia features in Aruba networks. Voice-aware scanning causes the AP to inhibit off-channel scanning when a multimedia call is active, returning to normal scanning operations when the call terminates. Similarly, potentially disruptive activity such as 802.1X re-keying is automatically suspended during FaceTime calls to the Apple device.

## Battery life

As we expect mobile devices to deliver more and more complex services, designers face a continual struggle to keep battery life at acceptable levels. Larger displays and more powerful processors make significant, and growing demands on the battery, but fortunately the WLAN chip is becoming more frugal, as chip designers use fabrication and power-management techniques to make the chip more frugal, and as recent 802.11 power-saving techniques become more widely deployed. Aruba supports a number of functions that work to minimize battery draw when Apple devices use Wi-Fi. A few are listed below:

- 802.11 power-save is the longest-standing standard technique. It allows mobile devices to switch off their radios and sleep while the AP buffers downlink frames, delivering them following a periodic DTIM beacon for which the client must wake. It is possible to extend the DTIM interval up to maybe give beacons, but data application performance is affected, as the round-trip delay for data sessions increases for a marginal improvement in battery life. Aruba normally recommends a DTIM interval of 2-4 beacons. 802.11 power-save helps 'idle' battery life, but is not a factor during a FaceTime or voice-over-Wi-Fi call. Apple devices support and use 802.11 power-save.

- WMM-PS (taken from 802.11 U-APSD) is very effective in extending battery life during multimedia calls such as FaceTime when many frames per second are exchanged. It allows the AP to buffer downlink frames until the client device transmits, thereby letting the client switch off its radio until its next transmission, rather than remaining constantly in receive mode in case a downlink frame arrives.Aruba's measurements indicate that WMM-PS can reduce on-call WLAN chip power usage by a factor of 2-3 in most cases. While all Aruba WLAN equipment is WMM-PS capable, Apple devices, despite WMM-PS support in the WLAN chip, do not currently use the feature. We hope this will be rectified in the near future.
- Since most of the WLAN chip power draw is associated with receiving and transmitting frames over-the-air, it is important to minimize extraneous traffic to the mobile device. Aruba has a number of features that are useful with Apple devices. The ARP proxy feature allows the AP to respond on the device's behalf to ARP queries on the LAN, shielding the device from unnecessary traffic.
- Similarly, discovery traffic associated with protocols such as VRRP can be blocked. Apple devices use a protocol called Bonjour to discover iTunes and printer instances in the environment: Bonjour uses MDNS multicast traffic, and is quite 'chatty.' It is possible to block this protocol on an Aruba WLAN, but as this disables a number of useful Apple features, network engineers should weigh the costs in utility against the gains in battery life and WLAN capacity.

| iPhone function | Power consumption when 'idle' (mW) | Power consumption during FaceTime (mW) | % of overall power draw, idle/FaceTime |
|---|---|---|---|
| **WALN chip** | 20 | 180 | 40/18% |
| **CPU and software** | 15 | 90 | 30/9% |
| **Display** | 0 | 700 | 0/70% |
| **Cellular chip** | 15 | 30 | 30/3% |
| **Total power draw** | **50** | **1000** | **100/100%** |

It is difficult to obtain completely accurate figures for Apple devices' power consumption: The figures above are extrapolated from various Aruba tests and give an indication of where the power is used. For context, the iPhone 4's battery is specified for 1420 mAh @ 3.7V for a nominal capacity of 5.25Wh, and a usable capacity of probably 4Wh. Our figures would predict a battery life of 80 hours idle and 4 hours on FaceTime (with no other applications active), which agrees reasonably well with practical observation.

Also note that driving the display places by far the greatest claim on battery life, about five times the Wi-Fi subsystem, even when on a FaceTime call. Nevertheless, we estimate that if WMM-PS and some of the other features mentioned above were implemented for FaceTime, iPhone battery life could be extended by about 5-10%, for the extreme scenario of continuous FaceTime calling.

## Conclusion

Video calling will become an increasingly widespread form of communication in coming years, moving from broad adoption in the consumer market to selective endorsement in the corporate world. The same executives who video-call their families when on the road will also use FaceTime when interacting with distant colleagues at work.

Interactive video stresses the corporate network, and the WLAN in particular, as it is a high bandwidth service that requires good quality of service. And as for all mobile devices, iPhones have limited battery capacity, so measures should be taken were possible to extend battery life.

Apple's FaceTime is based on standards such as STUN and H.264, but also includes unique features. Aruba has studied FaceTime and developed new features, notably device fingerprinting and application fingerprinting, to recognize and prioritize FaceTime traffic for good performance on the WLAN.

In this paper we showed how these features work to optimize FaceTime in corporate applications, and we also provided data to allow the network engineer to tune and control the enterprise network for this harbinger of interactive video services to come.

## References

1. Zeus Kerravala blogging on the No Jitter site about video communication in enterprises, http://www.nojitter.com/blog/228200977
2. Lucy Kellaway writing in the Economist's 'The World in 2011' on the rapid ascendance of corporate video communication, http://www.economist.com/node/17493438?story_id=17493438
3. FaceTime application calling instructions from Apple, http://support.apple.com/kb/HT4319
4. Apple well-known ports & protocols list, http://support.apple.com/kb/ts1629

## About Aruba Networks, Inc.

Aruba Networks is a leading provider of next-generation network access solutions for the mobile enterprise. The company's Mobile Virtual Enterprise (MOVE) architecture unifies wired and wireless network infrastructures into one seamless access solution for corporate headquarters, mobile business professionals, remote workers and guests. This unified approach to access networks enables IT organizations and users to securely address the Bring Your Own Device (BYOD) phenomenon, dramatically improving productivity and lowering capital and operational costs.

Listed on the NASDAQ and Russell 2000® Index, Aruba is based in Sunnyvale, California, and has operations throughout the Americas, Europe, Middle East, and Asia Pacific regions. To learn more, visit Aruba at http://www.arubanetworks.com. For real-time news updates follow Aruba on Twitter and Facebook.

ARUBA® networks

www.arubanetworks.com

1344 Crossman Avenue. Sunnyvale, CA 94089
1-866-55-ARUBA | Tel. +1 408.227.4500 | Fax. +1 408.227.4550 | info@arubanetworks.com