SiliconServer Data Sharing and Security White Paper

BLUEARC™

Silicon Servers for the Optical Age

Executive Summary

The ability of enterprises to access UNIX® and Windows NT® stored data both separately and from a shared common pool is growing in importance. Although Windows NT has advanced into business sectors that were the previous preserve of UNIX, the latter is holding strong in many of the sectors in which it originally found favor. Additionally, Linux® now offers a robust and secure alternative to Windows for many users.

A good example of the need to use a shared pool of data is the CAE (Computer Aided Engineering) sector, in which networked groups need to access common data in a secure environment. Because UNIX, with its networking strength, has always proved popular with CAE users, it continues to be used for scaled applications. However, at the same time, design houses are inclined to take advantage of Windows for reasons of cost and versatility. The result is that separate UNIX and Windows libraries are maintained in certain design houses, containing copies of the same files. This creates the need for a commonly shared independent file system.

BlueArc recognizes that, while a particular operating system is best suited to a particular environment, and an enterprise needs to be able to leverage the optimum benefits of all the operating systems with which they work individually, secure access to common data shared between UNIX, Linux and Windows is a key business advantage.

In the development of its SiliconServer Storage System, BlueArc has placed considerable emphasis on providing the data access and security traditionally associated with each native operating system. The SiliconServer file system provides enhanced combined security for UNIX and Windows clients, with integrated file locking, accurate translation of access credentials and per-file permissions, and secure user/user group mapping.

The SiliconServer Storage System has been designed to make using a shared common pool of storage as seamless and as native as possible from UNIX and Windows environments.

Limitations of Traditional Storage Solutions

UNIX and Windows server and storage solutions are traditionally isolated. The security models on each system have developed independently and there is a traditional mistrust about the two environments sharing a common pool of data. Existing Windows/UNIX file sharing implementations, including those implemented on other Network Attached Storage (NAS) solutions and file sharing applications, have significant limitations:

File System Integration

Current file system implementations start with the premise that a file system is in either UNIX or Windows format and that the files must have either UNIX or Windows attributes. This approach introduces an undesirable level of complexity when sharing data between UNIX and Windows: files that need to be shared are split between volumes of UNIX files that need to be shared with Windows users, and volumes of Windows files that need to be shared with UNIX. The resultant volumes or artificial sub-groupings of shared files bring in extra rules and process operations that are not native to either the Windows or UNIX environments. It is evident that a common file system is required, with all files capable of carrying both UNIX and Windows attributes, so that the non-native grouping, and the administrative overhead of managing groups, can be avoided completely.

Common File Locking

Because file sharing is often grafted on top of a UNIX or Windows NT kernel, certain file locking modes that are non-native to the host kernel cannot be implemented correctly or efficiently. This leads to a potential problem with file access through lack of a secure, integrated locking mechanism.

Security Mapping

The translation of a UNIX user identity to the NT security model, and of the UNIX file "Mode" to an NT Access Control List and vice versa, seems to be a hard task for some existing file sharing products, which have made various compromises in the accuracy and the performance of the translations. Accurate security mapping is essential if a user has both a Windows and a UNIX client account to access networked storage, yet some implementations lose detail in the translation between the UNIX and Windows environments. This often leads to artificial access restrictions in one environment that do not exist in the other. It can also lead to a security hole.

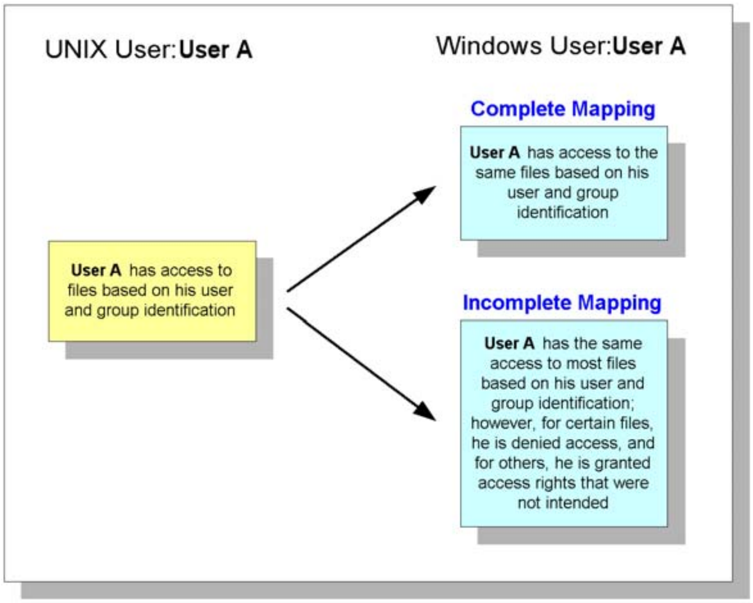
UNIX User: User A
Windows User: User A
User A has access to files based on his user and group identification
Complete Mapping
User A has access to the same files based on his user and group identification
Incomplete Mapping
User A has the same access to most files based on his user and group identification; however, for certain files, he is denied access, and for others, he is granted access rights that were not intended

Figure 1: UNIX/Windows file permissions mapping example, highlighting the effect of incomplete security mapping

This white paper discusses the ways in which the BlueArc SiliconServer Storage System addresses each of the above limitations to deliver fully integrated Windows and UNIX secured access to stored files.

The SiliconServer Storage System

The SiliconServer Storage System supports the following protocols: NFS (Network File System), CIFS (Common Internet File System), and FTP (File Transfer Protocol).

The architecture is based on a common file system and common integrated lock manager. The common file system carries UNIX and Windows attributes as metadata attached to each file, thereby offering a significantly improved UNIX/Windows file sharing solution, and the application of simultaneous enhanced security mapping between the UNIX and Windows security models.

In order for files created via NFS to be used by CIFS clients and vice versa, the SiliconServer understands the correspondence between UNIX and Windows users. A file's security information is held purely in native format until the non-native environment requests it, so this configuration is only required for users who share files between the UNIX and Windows environments. When operating on files that are exported to UNIX clients only (i.e. that are not shared by Windows clients), the SiliconServer does no mapping to the NT security model and the files retain purely UNIX security information. Equally, UNIX security configuration plays no role in serving files that are solely shared between Windows clients. In this way, where inter-environment file sharing is not required, it is perfectly straightforward to use the SiliconServer as a CIFS server and as a NFS server at the same time on two distinct groups of files without any need for configuration and without any concern that the security models may interfere in any way.

The following sections describe the operation of CIFS and NFS protocols both when interoperating and when operating only on their native files.

The SiliconServer - Full Integration of UNIX and Windows files

The SiliconServer provides fully integrated file system security through the seamless use of CIFS and NFS, without the use of sub-volumes, and without any performance compromises.

Files on the SiliconServer Storage System can contain UNIX only, NT only, or both UNIX and NT, data and security information. This is important in integrated (mixed) security mode operation, as will become clear in the following description of supported sharing modes on the SiliconServer Storage System.

CIFS Access to Native CIFS Files

When CIFS clients access “native” files on the system - “native” in this instance means files with Windows security information - security information on the user needs to be checked against per-file security information in order to determine whether or not an operation is permitted.

Security information on a user is contained in the Access Token, comprising the user Security Identifier (SID), Primary Group SID, and other SIDs. It is obtained from the domain controller when the user is authenticated and is cached locally in the SiliconServer for use throughout the duration of the user’s session.

The per-file security information is contained in a file’s Security Descriptor. The descriptor comprises an Owner SID, Group SID and the Access Control List (ACL) for the file. The ACL can contain a list of Access Control Entries that can richly define “access allowed” and “access denied”. Security properties for individual files on disk are configurable from the client using the client’s standard built-in file security mechanisms.

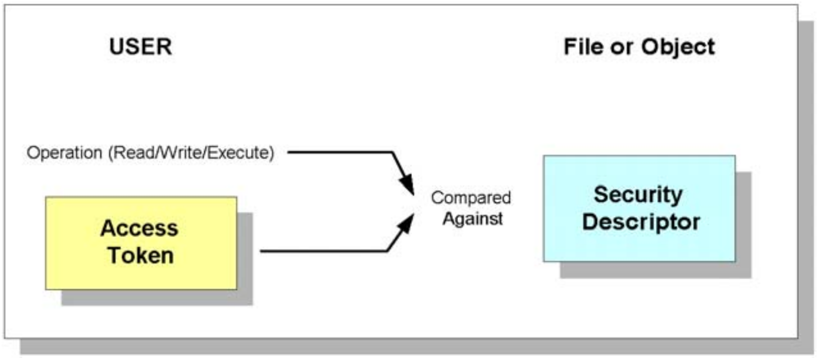

Figure 2: CIFS Native Access to Stored Files

By adhering to CIFS standards for permission checking, the SiliconServer fits seamlessly into the NT security model. Once the SiliconServer is configured with basic domain and domain controller information, no additional configuration is required.

CIFS Access to Non-Native CIFS Files

The SiliconServer has comprehensive capabilities for working with UNIX files, with accurate translation of UNIX file properties and UNIX access credentials. It provides the following functionality to support CIFS client access to non-native files ("non-native" in this instance means files without Windows security information):

- ❑ Configurable User and Group Mapping tables for mapping Windows user and group names to UNIX users and group names
- ❑ Translation of a Windows user Access Token to UNIX credentials for accessing UNIX files.
- ❑ Translation of UNIX file security attributes (Mode) to Windows attributes (Security Descriptor)
- ❑ Maintaining matching UNIX and Windows file attributes for each of the shared files on disk

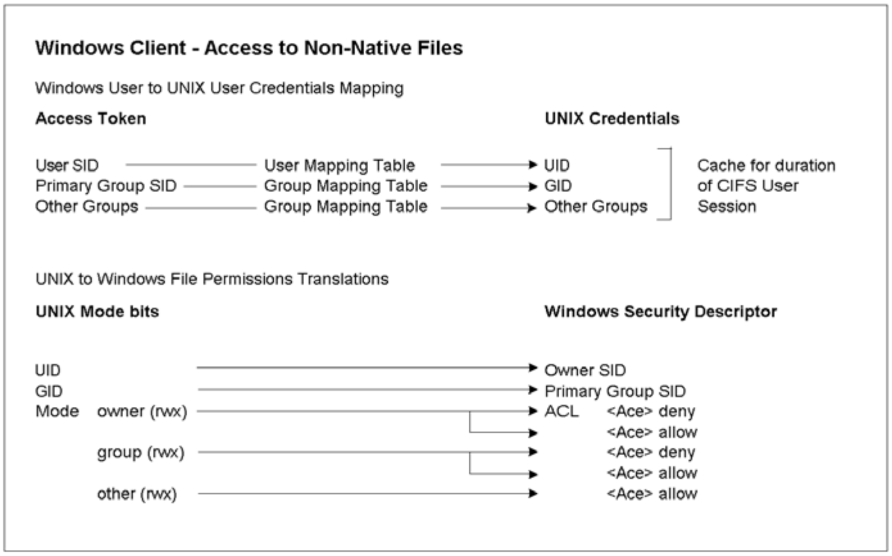

Figure 3: CIFS Access to Non-Native Stored Files

File access

When a CIFS client accesses a file that only has UNIX security information, the CIFS user making the request is mapped to a UNIX name. UNIX credentials for the user name are converted from the user's Access Token and the result is cached for the duration of the CIFS user session. The user's UNIX credentials are then used in accessing the UNIX file, as per normal NFS access - i.e. the UNIX credentials are compared with the file Mode security to determine whether the operation is allowed.

File properties of UNIX files when accessed from CIFS clients

The SiliconServer provides a significant improvement in representing security attributes for files shared between Windows and UNIX. The BlueArc implementation delivers a rich conversion of Mode information to NT Security Descriptor, by representing the Mode with a complete set of Access Control Entries (*owner deny/allow*, *group deny/allow* and *everyone allow* statuses). This is important. Without such a full conversion, group or user permissions may not be correctly reflected, leading to user access that is either more lax or more restrictive than the access dictated by the UNIX Mode of the file.

Another important difference is that the SiliconServer stores both UNIX and NT security attributes against a file. When the above Mode to Security Descriptor conversion is carried out, the result is stored in the file metadata, making the file native to both NFS and CIFS clients. Subsequently, any accesses to the file are treated as native accesses, for both NFS and CIFS protocols. Further, changes made to UNIX properties by UNIX clients are also made to the NT properties, and vice versa, so that the Security Descriptor and Mode of a file always accurately reflect the same access control for both NFS and CIFS clients.

The result is that there is fully preserved security, without any performance downside, when CIFS clients access shared files created by UNIX clients.

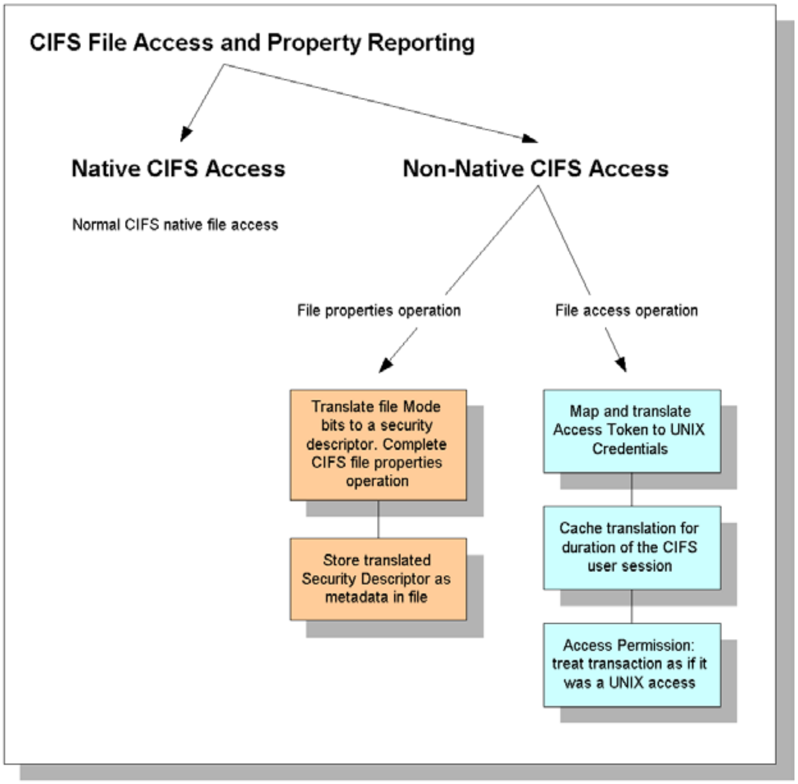
CIFS File Access and Property Reporting
Native CIFS Access
Non-Native CIFS Access
Normal CIFS native file access
File properties operation
File access operation
Translate file Mode bits to a security descriptor. Complete CIFS file properties operation
Store translated Security Descriptor as metadata in file
Map and translate Access Token to UNIX Credentials
Cache translation for duration of the CIFS user session
Access Permission: treat transaction as if it was a UNIX access

Figure 4 Flow diagram for a CIFS Non-Native file operation

NFS Access to Native NFS Files

When NFS clients access native files on the system – “native” in this instance means files with UNIX security information - the UNIX credentials supplied with the request are checked against the per-file security information (UNIX Mode - User ID, Group ID and **r**ead, **w**rite and e**x**ecute permissions) to determine whether or not an operation is permitted.

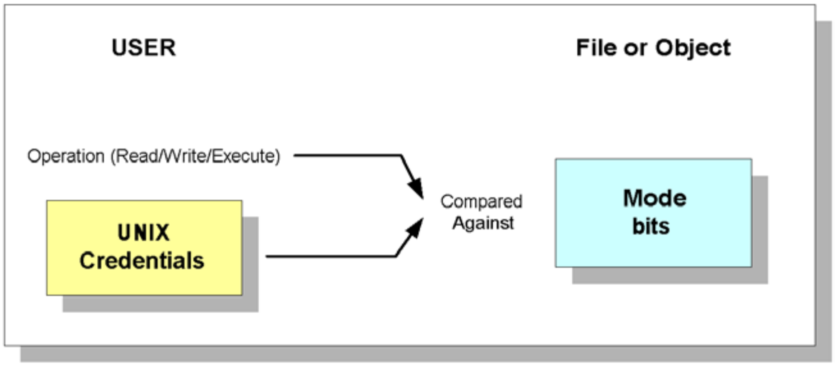

Figure 5: NFS native access to stored files

NFS Access to Non-Native NFS Files

The SiliconServer provides the following functionality to support NFS client access to non-native NFS files (“non-native” in this instance means files without UNIX security information):

- ❑ Configurable User and Group Mapping tables for mapping UNIX users and group names to Windows user and group names.
- ❑ Translation of UNIX credentials to a Windows user Access Token for accessing Windows files.
- ❑ Translation of Windows attributes (Security Descriptor) to UNIX file security attributes (Mode).
- ❑ Maintaining matching Windows and UNIX file attributes for each of the shared files on disk.

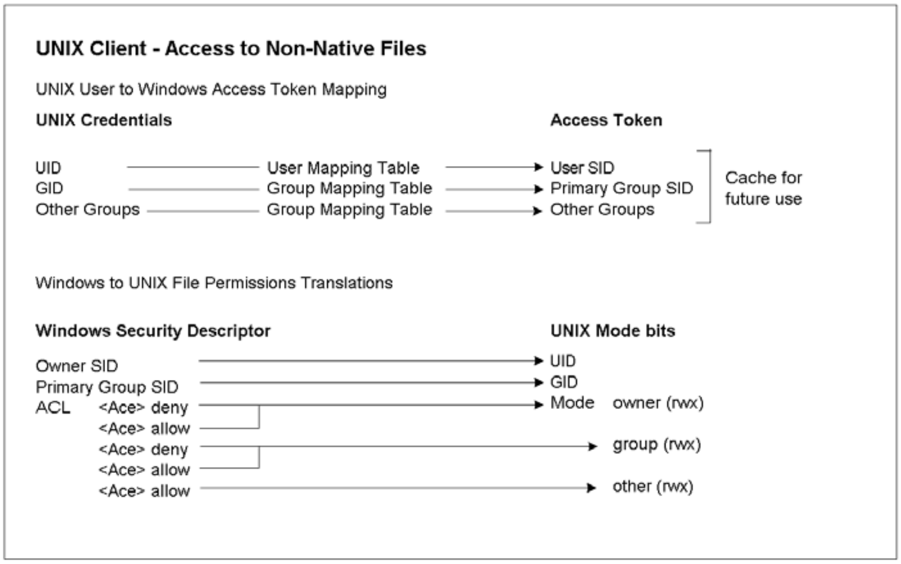

Figure 6: NFS access to Non-Native stored files

File access

When an NFS client accesses a file that only has Windows security information, the NFS user making the request is mapped to a Windows name. UNIX credentials for the user name are mapped and converted to create an NT Access Token and the result is held in a cache of recent conversions. The user's Access Token is then used in accessing the Windows file, as per normal CIFS access - i.e. the Access Token is checked against per-file security information (security descriptor) to determine whether the operation is allowed.

File properties of Windows files when accessed from NFS clients

The SiliconServer provides a significant improvement in representing security attributes for files shared between UNIX and Windows. The BlueArc implementation delivers a rich conversion of NT Security Descriptor to Mode. Without such a full conversion, group or user permissions may not be correctly reflected, leading to user access that is either more lax or more restrictive than the access dictated by the file's Windows ACL.

Another important difference is that the SiliconServer stores both UNIX and NT security attributes associated with a file. When the Security Descriptor to Mode conversion is carried out, the result is stored in the file metadata, making the file subsequently native to NFS clients. Further, changes made to Windows properties by Windows clients are also made to the UNIX properties, and vice versa, so that the Mode and Security Descriptor of a file always accurately reflect the same access control for both CIFS and NFS clients.

The result is that there is fully preserved security, without any performance downside, when NFS clients access shared files created by Windows clients.

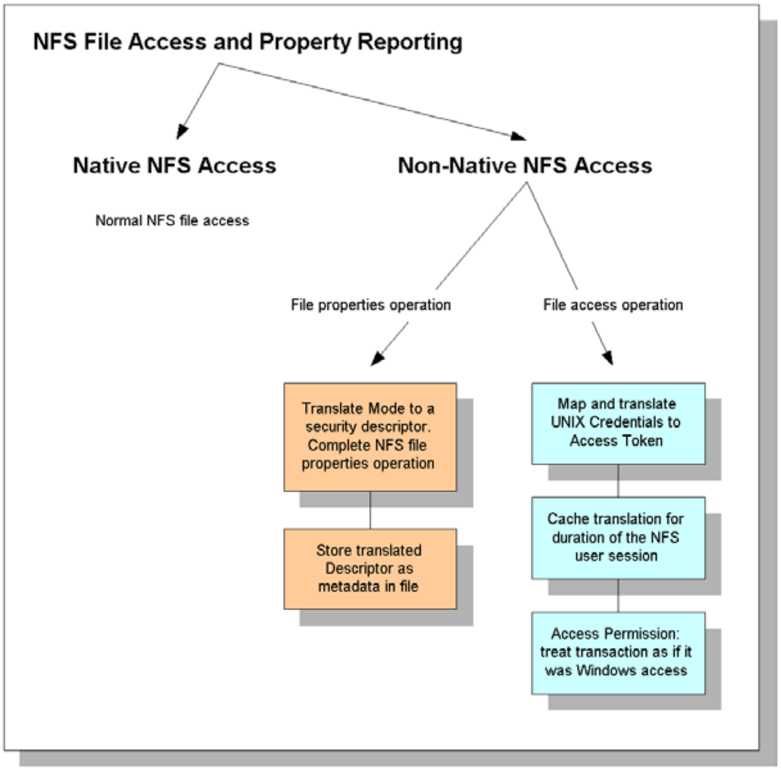
NFS File Access and Property Reporting
Native NFS Access
Normal NFS file access
Non-Native NFS Access
File properties operation
File access operation
Translate Mode to a security descriptor. Complete NFS file properties operation
Store translated Descriptor as metadata in file
Map and translate UNIX Credentials to Access Token
Cache translation for duration of the NFS user session
Access Permission: treat transaction as if it was Windows access

Figure 7: Flow diagram for a CIFS Non-native file operation

Common File Locking

The server provides full inter-working between NFS and Windows NT file locking - once a file range is locked by an NFS client, the lock is respected by a CIFS client, and vice versa.

This is to prevent multiple clients (whether they be UNIX or Windows) from attempting to write to the same section of a file simultaneously.

BlueArc's use of a common integrated lock manager means that there is no possibility of unsynchronized access to data, which could result in files becoming corrupted by simultaneous UNIX and Windows write activity.

Follow-on Benefits

To support the integrated security architecture of the SiliconServer, files stored within the server arrays carry full UNIX attributes, full Windows NT attributes, or both. This is also beneficial in the backup process. The SiliconServer NDMP (Network Data Management Protocol) implementation can be used to back up and restore a mixture of files created by a variety of operating systems, via a single, standard NDMP client backup management application, which can be running under either Windows or UNIX. This avoids the cost and overhead of supporting a separate backup solution for UNIX and Windows.

Additionally, the file security attributes for both Windows and UNIX are simultaneously backed up as part of the standard process. This alleviates the need to rebuild a proprietary rules-based security model when restoring file data from a backup.

Conclusion

The SiliconServer Storage System provides a solid, secure, and always available, Network Attached Storage system for both UNIX native and Windows native environments, and, regardless of the operating system, provides a business service that is both dependable and continuous.

The server provides seamless, integrated security for mixed operating system environments, with the following additional benefits:

- ❑ The lack of “rules” (e.g. the avoidance of different, artificial sub-volumes to support UNIX and Windows environments) means no potential for file access errors and reduced administration costs, compared with traditional file sharing implementations.
- ❑ The common file locking enables the provision of a central pool of secure, accessible storage from both Windows and UNIX, which can be shared without synchronicity problems.
- ❑ The SiliconServer maps security across UNIX and Windows with no incongruities. UNIX, Linux and Windows clients accessing a common pool of storage are given the access rights they are expected to have - no more, no less.

The SiliconServer thus breaks the mould of compromise that has been prevalent in mixed mode Network Attached Storage systems up until now, offering true integration in a single, common file system, with the mapping of both users and user groups to and from NFS to Windows NT, and common file locking.

General Information

US: info@bluearc.com

UK: uk_info@bluearc.com

Marketing Inquiries

marketing@bluearc.com

Sales Inquiries

US: sales@bluearc.com

UK: sales_uk@bluearc.com

References

BlueArc's SiliconServer™ white paper

BlueArc's FTP Support white paper

BlueArc's High Availability white paper

Copyright

The following are trademarks licensed to BlueArc Corporation, registered in the USA and other countries:

BlueArc™

The BlueArc™ logo

the SiliconServer Storage System™

the SiliconServer™ architecture upon which the SiliconServer Storage System™ is based.

Contact Details

For more information, contact BlueArc at:

BlueArc Corporation	BlueArc UK Ltd.
Corporate Headquarters	Queensgate House
225 Baypointe Parkway	Cookham Road
San Jose, CA 95134	Bracknell RG12 1RB
USA	United Kingdom
T 408 576 6600	T +44 (0) 1344 408 200
F 408 576 6601	F +44 (0) 1344 408 202

Or visit our website at: **www.bluearc.com**