

BIG-IP[®] System: External Cryptographic Server Offload Implementation

Version 11.6



Table of Contents

Legal Notices.....	5
Acknowledgments.....	7
Chapter 1: Implementing External Cryptographic Server Offload with BIG-IP	
systems.....	13
Overview: Implementing external cryptographic server offload.....	14
Creating a Client SSL profile on a client BIG-IP system.....	15
Creating a pool on a client BIG-IP system.....	15
Creating a virtual server on a client BIG-IP system.....	15
Creating a Server SSL profile on a client BIG-IP system.....	16
Creating a crypto client object on a client BIG-IP system.....	16
Creating a Client SSL profile on a server BIG-IP system.....	17
Creating a crypto server object on a server BIG-IP system.....	17
Verifying the crypto client and crypto server.....	17

Legal Notices

Publication Date

This document was published on August 20, 2014.

Publication Number

MAN-0546-00

Copyright

Copyright © 2014, F5 Networks, Inc. All rights reserved.

F5 Networks, Inc. (F5) believes the information it furnishes to be accurate and reliable. However, F5 assumes no responsibility for the use of this information, nor any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent, copyright, or other intellectual property right of F5 except as specifically described by applicable user licenses. F5 reserves the right to change specifications at any time without notice.

Trademarks

AAM, Access Policy Manager, Advanced Client Authentication, Advanced Firewall Manager, Advanced Routing, AFM, Application Acceleration Manager, Application Security Manager, APM, ARX, AskF5, ASM, BIG-IP, BIG-IQ, Cloud Extender, CloudFucious, Cloud Manager, Clustered Multiprocessing, CMP, COHESION, Data Manager, DevCentral, DevCentral [DESIGN], DNS Express, DSC, DSI, Edge Client, Edge Gateway, Edge Portal, ELEVATE, EM, Enterprise Manager, ENGAGE, F5, F5 [DESIGN], F5 Certified [DESIGN], F5 Networks, F5 SalesXchange [DESIGN], F5 Synthesis, f5 Synthesis, F5 Synthesis [DESIGN], F5 TechXchange [DESIGN], Fast Application Proxy, Fast Cache, FirePass, Global Traffic Manager, GTM, GUARDIAN, iApps, IBR, iCall, Intelligent Browser Referencing, Intelligent Compression, IPv6 Gateway, iControl, iHealth, iQuery, iRules, iRules OnDemand, iSession, L7 Rate Shaping, LC, Link Controller, LineRate, LineRate Systems [DESIGN], Local Traffic Manager, LROS, LTM, Message Security Manager, MobileSafe, MSM, OneConnect, Packet Velocity, PEM, Policy Enforcement Manager, Protocol Security Manager, PSM, Real Traffic Policy Builder, SalesXchange, ScaleN, SDAC (except in Japan), SDC, Signalling Delivery Controller, Solutions for an application world, Software Designed Applications Services, SSL Acceleration, StrongBox, SuperVIP, SYN Check, TCP Express, TDR, TechXchange, TMOS, TotALL, Traffic Management Operating System, Traffix (except Germany), Traffix [DESIGN] (except Germany), Transparent Data Reduction, UNITY, VAULT, vCMP, VE F5 [DESIGN], Versafe, Versafe [DESIGN], VIPRION, Virtual Clustered Multiprocessing, WebSafe, and ZoneRunner, are trademarks or service marks of F5 Networks, Inc., in the U.S. and other countries, and may not be used without F5's express written consent.

All other product and company names herein may be trademarks of their respective owners.

Patents

This product may be protected by one or more patents indicated at:
<http://www.f5.com/about/guidelines-policies/patents>

Export Regulation Notice

This product may include cryptographic software. Under the Export Administration Act, the United States government may consider it a criminal offense to export this product from the United States.

RF Interference Warning

This is a Class A product. In a domestic environment this product may cause radio interference, in which case the user may be required to take adequate measures.

FCC Compliance

This equipment has been tested and found to comply with the limits for a Class A digital device pursuant to Part 15 of FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This unit generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user, at his own expense, will be required to take whatever measures may be required to correct the interference.

Any modifications to this device, unless expressly approved by the manufacturer, can void the user's authority to operate this equipment under part 15 of the FCC rules.

Canadian Regulatory Compliance

This Class A digital apparatus complies with Canadian ICES-003.

Standards Compliance

This product conforms to the IEC, European Union, ANSI/UL and Canadian CSA standards applicable to Information Technology products at the time of manufacture.

Acknowledgments

This product includes software developed by Bill Paul.

This product includes software developed by Jonathan Stone.

This product includes software developed by Manuel Bouyer.

This product includes software developed by Paul Richards.

This product includes software developed by the NetBSD Foundation, Inc. and its contributors.

This product includes software developed by the Politecnico di Torino, and its contributors.

This product includes software developed by the Swedish Institute of Computer Science and its contributors.

This product includes software developed by the University of California, Berkeley and its contributors.

This product includes software developed by the Computer Systems Engineering Group at the Lawrence Berkeley Laboratory.

This product includes software developed by Christopher G. Demetriou for the NetBSD Project.

This product includes software developed by Adam Glass.

This product includes software developed by Christian E. Hopps.

This product includes software developed by Dean Huxley.

This product includes software developed by John Kohl.

This product includes software developed by Paul Kranenburg.

This product includes software developed by Terrence R. Lambert.

This product includes software developed by Philip A. Nelson.

This product includes software developed by Herb Peyerl.

This product includes software developed by Jochen Pohl for the NetBSD Project.

This product includes software developed by Chris Provenzano.

This product includes software developed by Theo de Raadt.

This product includes software developed by David Muir Sharnoff.

This product includes software developed by SigmaSoft, Th. Lockert.

This product includes software developed for the NetBSD Project by Jason R. Thorpe.

This product includes software developed by Jason R. Thorpe for And Communications, <http://www.and.com>.

This product includes software developed for the NetBSD Project by Frank Van der Linden.

This product includes software developed for the NetBSD Project by John M. Vinopal.

This product includes software developed by Christos Zoulas.

This product includes software developed by the University of Vermont and State Agricultural College and Garrett A. Wollman.

This product includes software developed by Balazs Scheidler (bazsi@balabit.hu), which is protected under the GNU Public License.

This product includes software developed by Niels Mueller (nisse@lysator.liu.se), which is protected under the GNU Public License.

Acknowledgments

In the following statement, "This software" refers to the Mitsumi CD-ROM driver: This software was developed by Holger Veit and Brian Moore for use with 386BSD and similar operating systems. "Similar operating systems" includes mainly non-profit oriented systems for research and education, including but not restricted to NetBSD, FreeBSD, Mach (by CMU).

This product includes software developed by the Apache Group for use in the Apache HTTP server project (<http://www.apache.org/>).

This product includes software licensed from Richard H. Porter under the GNU Library General Public License (© 1998, Red Hat Software), www.gnu.org/copyleft/lgpl.html.

This product includes the standard version of Perl software licensed under the Perl Artistic License (© 1997, 1998 Tom Christiansen and Nathan Torkington). All rights reserved. You may find the most current standard version of Perl at <http://www.perl.com>.

This product includes software developed by Jared Minch.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>).

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com).

This product contains software based on oprofile, which is protected under the GNU Public License.

This product includes RRDtool software developed by Tobi Oetiker (<http://www.rrdtool.com/index.html>) and licensed under the GNU General Public License.

This product contains software licensed from Dr. Brian Gladman under the GNU General Public License (GPL).

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).

This product includes Hypersonic SQL.

This product contains software developed by the Regents of the University of California, Sun Microsystems, Inc., Scriptics Corporation, and others.

This product includes software developed by the Internet Software Consortium.

This product includes software developed by Nominum, Inc. (<http://www.nominum.com>).

This product contains software developed by Broadcom Corporation, which is protected under the GNU Public License.

This product contains software developed by MaxMind LLC, and is protected under the GNU Lesser General Public License, as published by the Free Software Foundation.

This product includes unbound software from NLnetLabs. Copyright ©2007. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of NLnetLabs nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS

INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes Intel QuickAssist kernel module, library, and headers software licensed under the GNU General Public License (GPL).

This product includes software licensed from Gerald Combs (gerald@wireshark.org) under the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or any later version. Copyright ©1998 Gerald Combs.

This product includes software developed by Thomas Williams and Colin Kelley. Copyright ©1986 - 1993, 1998, 2004, 2007

Permission to use, copy, and distribute this software and its documentation for any purpose with or without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. Permission to modify the software is granted, but not the right to distribute the complete modified source code. Modifications are to be distributed as patches to the released version. Permission to distribute binaries produced by compiling modified sources is granted, provided you

1. distribute the corresponding source modifications from the released version in the form of a patch file along with the binaries,
2. add special version identification to distinguish your version in addition to the base release version number,
3. provide your name and address as the primary contact for the support of your modified version, and
4. retain our contact information in regard to use of the base software.

Permission to distribute the released version of the source code along with corresponding source modifications in the form of a patch file is granted with same provisions 2 through 4 for binary distributions. This software is provided "as is" without express or implied warranty to the extent permitted by applicable law.

This product includes software developed by Brian Gladman, Worcester, UK Copyright ©1998-2010. All rights reserved. The redistribution and use of this software (with or without changes) is allowed without the payment of fees or royalties provided that:

- source code distributions include the above copyright notice, this list of conditions and the following disclaimer;
- binary distributions include the above copyright notice, this list of conditions and the following disclaimer in their documentation.

This software is provided "as is" with no explicit or implied warranties in respect of its operation, including, but not limited to, correctness and fitness for purpose.

This product includes software developed by the Computer Systems Engineering Group at Lawrence Berkeley Laboratory. Copyright ©1990-1994 Regents of the University of California. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment: This product includes software developed by the Computer Systems Engineering Group at Lawrence Berkeley Laboratory.
4. Neither the name of the University nor of the Laboratory may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes software developed by Sony Computer Science Laboratories Inc. Copyright © 1997-2003 Sony Computer Science Laboratories Inc. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY SONY CSL AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL SONY CSL OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product contains software developed by Google, Inc. Copyright ©2011 Google, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This product includes software developed by Ian Gulliver ©2006, which is protected under the GNU General Public License, as published by the Free Software Foundation.

This product includes software developed by Jeremy Ashkenas and DocumentCloud, and distributed under the MIT license. Copyright © 2010-2013 Jeremy Ashkenas, DocumentCloud.

This product includes gson software, distributed under the Apache License version 2.0. Copyright © 2008-2011 Google Inc.

This product includes crypto.js software, copyright © 2009-2013, Jeff Mott, and distributed under the BSD New license.

This product includes jxrlib software, copyright ©2009 Microsoft Corp. All rights reserved. Distributed under the new BSD license.

This product contains OpenLDAP software, which is distributed under the OpenLDAP v2.8 license (BSD3-like).

This product includes Racoon 2 software, copyright © 2003-2005 WIDE Project. All rights reserved. Distributed under a BSD-like license.

This product includes node-uuid software, copyright © 2010-2012, Robert Kieffer, and distributed under the MIT license.

This product includes opensv software, which is distributed under the Apache 2.0 license.

This product includes owasp-jave-encoder software, copyright © 2014, Jeff Ichnowski, and distributed under the New BSD license.

Chapter

1

Implementing External Cryptographic Server Offload with BIG-IP systems

- *Overview: Implementing external cryptographic server offload*

Overview: Implementing external cryptographic server offload

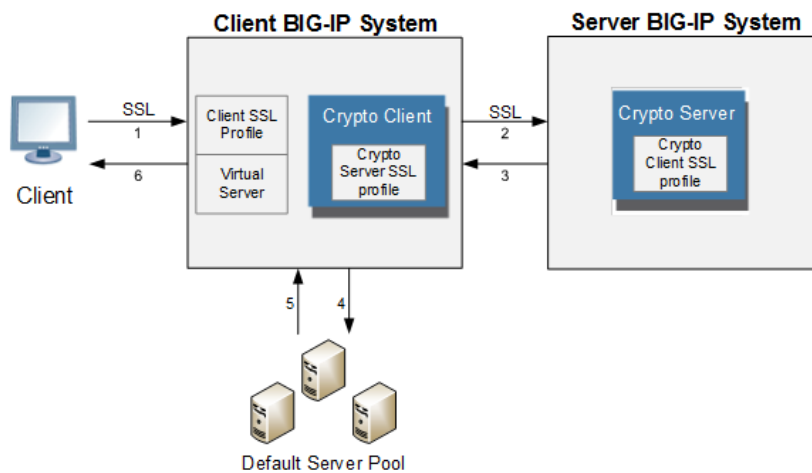
You can offload cryptographic operations to an external BIG-IP® system. For example, you can set up an LTM VE instance (the crypto client) to offload cryptographic operations, such as an RSA decryption operation for an SSL handshake, to an external BIG-IP system (the crypto server) that supports cryptographic hardware acceleration.

In general, the setup process includes configuring a client BIG-IP system as a crypto client and a server BIG-IP system as a crypto server, and ensures secure communication between the end user, the crypto client, and the crypto server.

Important: Both the crypto client and crypto server must be running BIG-IP software version 11.6.0 or later.

Important: Before you perform the tasks in this implementation, verify that each BIG-IP system has the default device certificate, `default.crt`, installed on it. For more information about device certificates, see *BIG-IP® Digital Certificates: Administration*.

This illustration depicts an external cryptographic offload configuration.



The illustration shows the BIG-IP configuration objects that are required for implementing the external cryptographic server offload feature, as well as the flow of client traffic that occurs. In the illustration, one BIG-IP system includes a virtual server configured with the destination IP address for application traffic coming from a client system. Because the client traffic uses SSL, the BIG-IP system with the virtual server must include a standard Client SSL profile, which causes cryptographic functions to be offloaded from the selected destination server (pool member) to that BIG-IP system.

Once this BIG-IP system has assumed cryptographic functions from the destination server, the BIG-IP system can offload these functions to another BIG-IP system to handle the actual cryptographic processing. To enable the BIG-IP system to offload the cryptographic processing to another BIG-IP system, you must designate the two BIG-IP systems as a crypto client and crypto server, and you must create an SSL profile on each system that is optimized for BIG-IP-to-BIG-IP cryptographic processing (a crypto-optimized Server SSL profile for the BIG-IP crypto client and crypto-optimized Client SSL profile for the BIG-IP crypto server).

Task summary

Creating a Client SSL profile on a client BIG-IP system

Creating a pool on a client BIG-IP system

Creating a virtual server on a client BIG-IP system
Creating a Server SSL profile on a client BIG-IP system
Creating a crypto client object on a client BIG-IP system
Creating a Client SSL profile on a server BIG-IP system
Creating a crypto server object on a server BIG-IP system
Verifying the crypto client and crypto server

Creating a Client SSL profile on a client BIG-IP system

You create a Client SSL profile on a client BIG-IP® system to authenticate and decrypt/encrypt client-side application traffic.

1. On the Main tab, click **Local Traffic > Profiles > SSL > Client**.
The Client profile list screen opens.
2. Click **Create**.
The New Client SSL Profile screen opens.
3. Configure all profile settings as needed.
4. Click **Finished**.

After you create the Client SSL profile, you assign the profile to a virtual server. The BIG-IP® system can apply SSL security to the type of application traffic for which the virtual server is configured to listen.

Creating a pool on a client BIG-IP system

You can create a pool of servers on a client BIG-IP® system that you can group together to receive and process traffic.

1. On the Main tab, click **Local Traffic > Pools**.
The Pool List screen opens.
2. Click **Create**.
The New Pool screen opens.
3. In the **Name** field, type a unique name for the pool.
4. Using the **New Members** setting, add each resource that you want to include in the pool:
 - a) In the **Node Name** field, type a name for the node portion of the pool member.
This step is optional.
 - b) In the **Address** field, type an IP address.
 - c) In the **Service Port** field, type a port number, or select a service name from the list.
 - d) In the **Priority** field, type a priority number.
This step is optional.
 - e) Click **Add**.
5. Click **Finished**.

Creating a virtual server on a client BIG-IP system

A virtual server represents a destination IP address for application traffic on a client BIG-IP® system.

1. On the Main tab, click **Local Traffic > Virtual Servers**.
The Virtual Server List screen opens.
2. Click the **Create** button.
The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. In the **Destination Address** field, type the IP address in CIDR format.
The supported format is address/prefix, where the prefix length is in bits. For example, an IPv4 address/prefix is 10.0.0.1 or 10.0.0.0/24, and an IPv6 address/prefix is ff01::0020/64 or 2001:ed8:77b5:2:10:10:100:42/64. When you use an IPv4 address without specifying a prefix, the BIG-IP® system automatically uses a /32 prefix.
5. In the **Service Port** field, type a port number or select a service name from the **Service Port** list.
6. In the Resources area of the screen, from the **Default Pool** list, select the relevant pool name.
7. For the **SSL Profile (Client)** setting, from the **Available** list, select the name of the Client SSL profile you previously created, and using the Move button, move the name to the **Selected** list.

Creating a Server SSL profile on a client BIG-IP system

With a Server SSL profile, a client BIG-IP® system can perform decryption and encryption for server-side SSL traffic.

1. On the Main tab, click **Local Traffic > Profiles > SSL > Server**.
The SSL Server profile list screen opens.
2. Click **Create**.
The New Server SSL Profile screen opens.
3. In the **Name** field, type a unique name for the profile.
4. Select **crypto-client-default-serverssl** in the **Parent Profile** list.
5. Modify the settings, as required.
6. Click **Finished**.

Creating a crypto client object on a client BIG-IP system

You can create a crypto client object to enable a BIG-IP® system to act as a crypto client for external cryptographic server offload.

1. On the Main tab, click **System > Crypto Offloading > Crypto Client**.
The Crypto Client screen displays a list of crypto clients configured on the system.
2. Click **Create**.
3. In the **Name** field, type a unique name for the crypto client object.
4. In the **Address** field, type the IP address of the crypto server that you want to use for the crypto server object.
5. In the **Service Port** field, type a port number, or select a service name from the list.
6. In the **TCP Profiles** field, select **tcp**.
7. For the **SSL Profiles** setting, select the Server SSL profile that you previously created.

Creating a Client SSL profile on a server BIG-IP system

You create a Client SSL profile on a server BIG-IP® system to authenticate and decrypt/encrypt application traffic from the client BIG-IP system.

1. On the Main tab, click **Local Traffic > Profiles > SSL > Client**.
The Client profile list screen opens.
2. Click **Create**.
The New Client SSL Profile screen opens.
3. Select **crypto-server-default-clientssl** in the **Parent Profile** list.
4. Configure all profile settings as needed.
5. Click **Finished**.

Creating a crypto server object on a server BIG-IP system

You can create a crypto server object to enable your BIG-IP® system to act as a crypto server for external cryptographic server offload.

1. On the Main tab, click **System > Crypto Offloading > Crypto Server**.
The Crypto Server screen displays a list of crypto servers configured on the system.
2. Click **Create**.
3. In the **Name** field, type a unique name for the crypto server object.
4. In the **Address** field, type the IP address you want to use for the crypto server object.
5. In the **Service Port** field, type a port number, or select a service name from the list.
6. In the **TCP Profiles** field, select **tcp**.
7. For the **SSL Profiles** setting, select the Client SSL profile that you previously created.
8. (Optional) Using the **Crypto Client List** setting, add the crypto clients that can access the crypto server:
 - a) In the **Address** field, type a crypto client self IP address.
 - b) Click **Add**.

Verifying the crypto client and crypto server

After the client and server BIG-IP® systems have processed traffic, you can use `tmsh` to verify that the crypto client and crypto server systems are functioning properly.

1. Open the Traffic Management Shell (`tmsh`).
`tmsh`
2. Verify that the crypto client is functioning.
`show sys crypto client <crypto_client_name>`

A summary similar to this example displays:

```
-----
Sys::Crypto Client: crypto_client_name
-----
Received Packets      2
Received Bytes       48
```

```
Transmitted Packets  2
Transmitted Bytes    40
```

3. Verify that the crypto server is functioning.

```
show sys crypto server <crypto_server_name>
```

A summary similar to this example displays:

```
-----
Sys::Crypto Server: crypto_server_name
-----
Received Packets      2
Received Bytes       40
Transmitted Packets  2
Transmitted Bytes    48
```

Index

B

BIG-IP software requirements
 crypto client [14](#)
 crypto server [14](#)

C

client BIG-IP systems
 BIG-IP software requirements [14](#)
 creating a Client SSL profile [15](#)
 creating a crypto client object [16](#)
 creating a pool [15](#)
 creating a Server SSL profile [16](#)
 creating a virtual server [15](#)

Client SSL profiles
 creating on a client BIG-IP system [15](#)
 creating on a server BIG-IP system [17](#)

crypto client objects
 creating on a client BIG-IP system [16](#)
 verifying [17](#)

crypto clients, See client BIG-IP systems.

crypto offload, See external cryptographic server offload.

crypto server objects
 creating on a server BIG-IP system [17](#)
 verifying [17](#)

crypto servers, See server BIG-IP systems.

E

external cryptographic server offload
 BIG-IP software requirements [14](#)

external cryptographic server offload (*continued*)
 implementation overview [14](#)

P

pools
 creating on a client BIG-IP system [15](#)

profiles
 creating a Server SSL profile on a client BIG-IP system
[16](#)

S

server BIG-IP systems
 BIG-IP software requirements [14](#)
 creating a Client SSL profile [17](#)
 creating a crypto server object [17](#)

SSL profiles
 creating on a client BIG-IP system [15](#)
 creating on a server BIG-IP system [17](#)

T

tmsh
 verifying the crypto client [17](#)
 verifying the crypto server [17](#)

V

virtual servers
 creating on a client BIG-IP system [15](#)

