BIG-IP® Access Policy Manager® and F5 Access for iOS v2.1.0

2.1.0

Table of Contents

Legal notices

Publication Date

This document was published on February 13, 2017.

Publication Number

MAN-0393-07

Copyright

Copyright © 2017, 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

For a current list of F5 trademarks and service marks, see *http://www.f5.com/about/guidelines-policies/trademarks/*.

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: *https://f5.com/about-us/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.

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 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 Daniel Stenberg. Copyright ©1996 - 2012, Daniel Stenberg, (daniel@haxx.se). All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

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 OF THIRD PARTY RIGHTS. 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.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

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 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 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 ec2-tools software, copyright © 2008, Amazon Web Services, and licensed under the Amazon Software License. A copy of the License is located at http://aws.amazon.com/asl/ .

This product includes Apache Ant software, distributed by the Apache Software Foundation under the Apache License, version 2.0.

This product includes Boost libraries, which are distributed under the Boost license (http://www.boost.org/LICENSE_1_0.txt).

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

This product includes software developed by Computing Services at Carnegie Mellon University (http://www.cmu.edu/computing/).

This product includes libmagic software, copyright © Ian F. Darwin 1986, 1987, 1989, 1990, 1991, 1992, 1994, 1995. Software written by Ian F. Darwin and others; maintained 1994- Christos Zoulas.

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

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

This product includes opencsv 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.

This product may include Intel SDD software subject to the following license; check your hardware specification for details.

1. LICENSE. This Software is licensed for use only in conjunction with Intel solid state drive (SSD) products. Use of the Software in conjunction with non-Intel SSD products is not licensed hereunder. Subject to the terms of this Agreement, Intel grants to You a nonexclusive, nontransferable, worldwide, fully paid-up license under Intel's copyrights to:

   - copy the Software onto a single computer or multiple computers for Your personal, noncommercial use; and
   - make appropriate back-up copies of the Software, for use in accordance with Section 1a) above.

   The Software may contain the software or other property of third party suppliers, some of which may be identified in, and licensed in accordance with, any enclosed "license.txt" file or other text or file.

   Except as expressly stated in this Agreement, no license or right is granted to You directly or by implication, inducement, estoppel or otherwise. Intel will have the right to inspect or have an independent auditor inspect Your relevant records to verify Your compliance with the terms and conditions of this Agreement.

2. RESTRICTIONS. You will not:

   a. copy, modify, rent, sell, distribute or transfer any part of the Software, and You agree to prevent unauthorized copying of the Software; and,
   b. reverse engineer, decompile, or disassemble the Software; and,
   c. sublicense or permit simultaneous use of the Software by more than one user; and,
   d. otherwise assign, sublicense, lease, or in any other way transfer or disclose Software to any third party, except as set forth herein; and,
   e. subject the Software, in whole or in part, to any license obligations of Open Source Software including without limitation combining or distributing the Software with Open Source Software in a manner that subjects the Software or any portion of the Software provided by Intel hereunder to any license obligations of such Open Source Software. "Open Source Software" means any software that requires as a condition of use, modification and/or distribution of such software that such software or other software incorporated into, derived from or distributed with such software:

      a. be disclosed or distributed in source code form; or
      b. be licensed by the user to third parties for the purpose of making and/or distributing derivative works; or
      c. be redistributable at no charge.

   Open Source Software includes, without limitation, software licensed or distributed under any of the following licenses or distribution models, or licenses or distribution models substantially similar to any of the following:

   a. GNU’s General Public License (GPL) or Lesser/Library GPL (LGPL),

b. the Artistic License (e.g., PERL),
c. the Mozilla Public License,
d. the Netscape Public License,
e. the Sun Community Source License (SCSL),
f. vi) the Sun Industry Source License (SISL),
g. vii) the Apache Software license, and
h. viii) the Common Public License (CPL).

3. OWNERSHIP OF SOFTWARE AND COPYRIGHTS. Title to all copies of the Software remains with Intel or its suppliers. The Software is copyrighted and protected by the laws of the United States and other countries, and international treaty provisions. You may not remove any copyright notices from the Software. Intel may make changes to the Software, or to materials referenced therein, at any time and without notice, but is not obligated to support or update the Software. Except as otherwise expressly provided, Intel grants no express or implied right or license under Intel patents, copyrights, trademarks, or other intellectual property rights.

4. Entire Agreement. This Agreement contains the complete and exclusive statement of the agreement between You and Intel and supersedes all proposals, oral or written, and all other communications relating to the subject matter of this Agreement. Only a written instrument duly executed by authorized representatives of Intel and You may modify this Agreement.

5. LIMITED MEDIA WARRANTY. If the Software has been delivered by Intel on physical media, Intel warrants the media to be free from material physical defects for a period of ninety (90) days after delivery by Intel. If such a defect is found, return the media to Intel for replacement or alternate delivery of the Software as Intel may select.

6. EXCLUSION OF OTHER WARRANTIES. EXCEPT AS PROVIDED ABOVE, THE SOFTWARE IS PROVIDED "AS IS" WITHOUT ANY EXPRESS OR IMPLIED WARRANTY OF ANY KIND, INCLUDING WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT, OR FITNESS FOR A PARTICULAR PURPOSE. Intel does not warrant or assume responsibility for any errors, the accuracy or completeness of any information, text, graphics, links or other materials contained within the Software.

7. LIMITATION OF LIABILITY. IN NO EVENT WILL INTEL OR ITS SUPPLIERS BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION OR LOST INFORMATION) ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE, EVEN IF INTEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME JURISDICTIONS PROHIBIT EXCLUSION OR LIMITATION OF LIABILITY FOR IMPLIED WARRANTIES OR CONSEQUENTIAL OR INCIDENTAL DAMAGES, SO THE ABOVE LIMITATION MAY NOT APPLY TO YOU. YOU MAY ALSO HAVE OTHER LEGAL RIGHTS THAT VARY FROM JURISDICTION TO JURISDICTION.

8. TERMINATION OF THIS AGREEMENT. Intel may terminate this Agreement at any time if You violate its terms. Upon termination, You will immediately destroy the Software or return all copies of the Software to Intel.

9. APPLICABLE LAWS. Claims arising under this Agreement will be governed by the laws of Delaware, excluding its principles of conflict of laws and the United Nations Convention on Contracts for the Sale of Goods. You may not export the Software in violation of applicable export laws and regulations. Intel is not obligated under any other agreements unless they are in writing and signed by an authorized representative of Intel.

10. GOVERNMENT RESTRICTED RIGHTS. The Software is provided with "RESTRICTED RIGHTS." Use, duplication, or disclosure by the Government is subject to restrictions as set forth in FAR52.227-14 and DFAR252.227-7013 et seq. or their successors. Use of the Software by the Government constitutes acknowledgment of Intel's proprietary rights therein. Contractor or Manufacturer is Intel Corporation, 2200 Mission College Blvd., Santa Clara, CA 95054.

This product includes zipzap software, which is distributed under the BSD license. Copyright © 2012, Pixelglow Software. All rights reserved.

Chapter 1

Configuring Access Policy Manager for F5 Access

Overview: F5 Access for macOS

What does F5 Access do for mobile devices?

F5 Access for mobile devices provides full network access through BIG-IP® Access Policy Manager®. With network access, users can run applications such as RDP, SSH, Citrix, VMware View, and other enterprise applications on their mobile devices.

For information about how to use F5 Access on your device, refer to the *F5 Access for iOS User Guide*.

F5 Access features include:

- N-factor authentication (at least two input fields, password and passcode) support
- User name and password, client certificate, and RSA SecurID support
- Multiple input field support
- Credential caching support
- Support for TouchID authentication, PIN, or a device password to make a connection, when using cached credentials
- Support for DNS address space for split-tunneling configurations
- Support for checking information from client devices
- Support for automatically launching applications on client devices
- Support for roaming between cellular and WiFi networks
- Landing URI support
- Logging support to report issues
- Support for private-side internal proxy servers. Public-side proxy servers are not currently supported.
- Support client certificate for DTLS tunnels and SSL tunnels
- Per-app VPN support
- Support for SAML 2.0 features in BIG-IP®Access Policy Manager®
- iOS widget support

About SAML support

F5 Access for iOS devices provides the following SAML support:

- Service provider-initiated access only, for example, APM acting as the service provider (SP)
- Web Logon mode only
- Single Log-Out (SLO): supported only when the logout action is initiated from the client

When you use as a client performing SP-initiated access, first connects to BIG-IP® Access Policy Manager® (APM®). Because there is no assertion, APM redirects the client to the IdP. The IdP then authenticates the user and redirects back to the SP with assertion. APM then accepts the assertion and establishes a VPN connection. You can then access back-end resources through .

You can configure a BIG-IP system by configuring APM as an SP. The access policy that is associated with the configuration assigns a SAML AAA resource followed by a Network Access Resource. For more information about SAML configurations, refer to the *BIG-IP® Access Policy Manager®: Authentication and Single Sign-On* guide.

About supported authentication types

F5 Access for iOS provides these authentication types:

Authentication type	Description
VPN On-Demand	Provides the following three options: • Username and password • Client certificate • Client certificate + username and password (no runtime prompt)
Regular Logon	Provides the following three options: • Username and password • Client certificate • Client certificate + username and password (no runtime prompt)
Per-App VPN	Per-app VPN requires authentication without user intervention. Therefore, only authentication methods that require no user intervention are supported. • Client certificate • Username/password + client certificate (username and password must be specified in VPN configuration)
Web Logon	Provides the following four options: • Username and password • Username/password + RSA + any other server-side checks • Client certificate • Client certificate + username and password (no runtime prompt)

About establishing VPN connections

The F5 Access application (app) for mobile devices provides users with two options to establish a VPN tunnel connection. A user can start a tunnel connection explicitly with the F5 Access application, or implicitly through the VPN On-Demand functionality.

For example, a connection can be configured to automatically trigger whenever a certain domain or host name pattern is matched.

VPN On-Demand considerations:

- When VPN On-Demand initiates a connection, user intervention is not allowed. For example, if a password is needed for authentication, but is not supplied in the configuration, the connection fails. Note that RSA authentication is not supported.
- VPN On-Demand supports only two authentication types. After you have imported the configuration profile, you can perform configurations to add additional credential authentication using the application.
- VPN On-Demand does not work if you enable Web Logon.

About pre-logon checks supported for iOS devices

Access Policy Manager® can check unique identifying information from an iOS client device. The supported session variables, which become populated with the iOS client device information, are gathered automatically, and can easily be combined with an LDAP or AD query to implement white-listing in a custom action to improve access context. This information allows Access Policy Manager to perform pre-logon sequence checks and actions based on information about the connecting device. Using such information, Access Policy Manager can perform the following tasks:

- Deny access if the iOS version is less than the required level.
- Log UDID information.

This example displays an access policy with a custom action of Device ID Check to check the device's UDID.

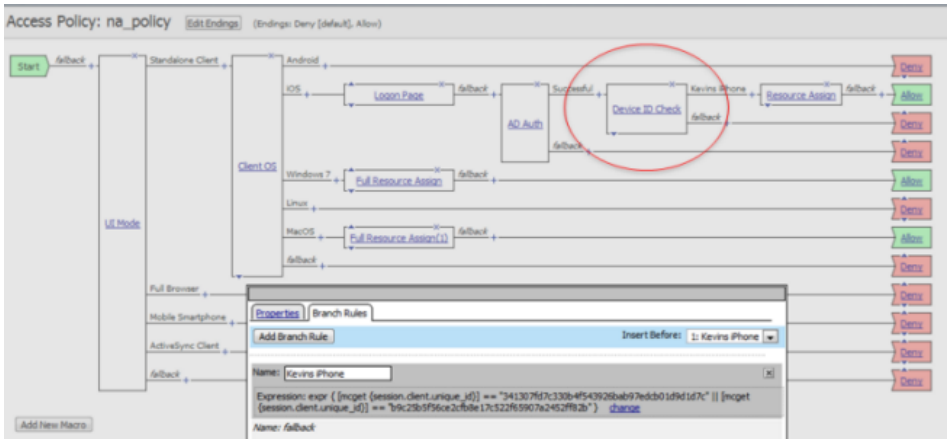


Figure 1: Example of a custom action for checking device's UDID

About automatically launching applications from mobile devices

You can configure F5 Access to launch an app with a registered URL scheme after a VPN connection is established.

Auto-launching applications from F5 Access

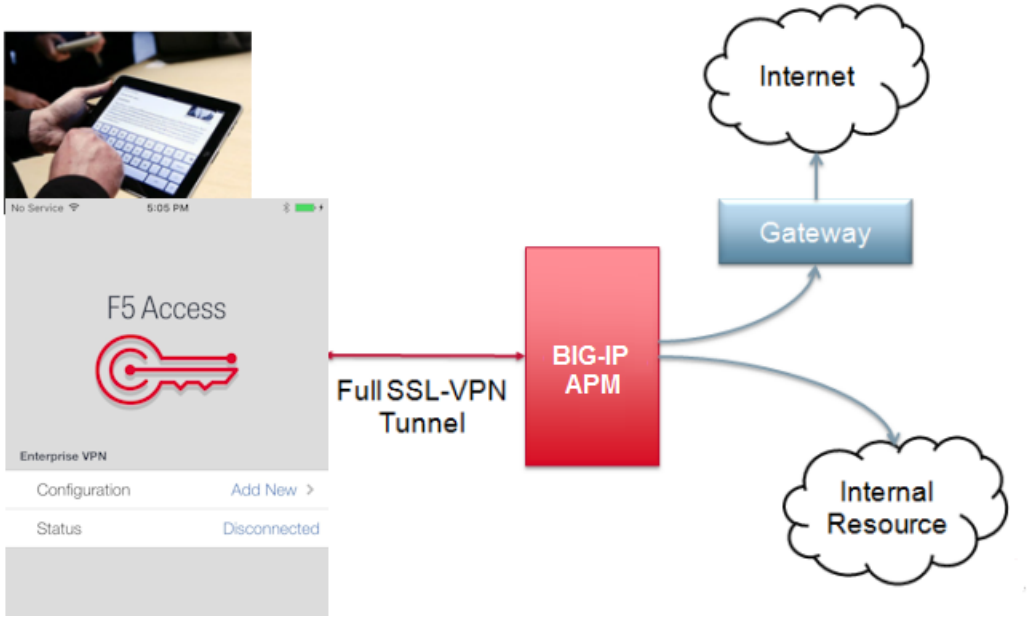
You can configure applications to automatically start on F5 Access once a connection is initiated.

1. On the Main tab, click **Access Policy**.
2. Navigate to **Network Access**, and select the name of your network access from the list.
3. Select the **Launch Applications** tab.

4. In the **Application Path** field, type in your application path in the form of a URL scheme, for example, `skype://14082734800?call`.
5. From the **Operating System** selection field, select your operating system type.
6. Click **Finished.**
   On the device itself, a pre-launch warning is issued before the local application executes.

About network integration on iOS devices

Access Policy Manager® provides web application-level security to prevent malware attacks. As an administrator, you can enforce all web access through a secured gateway, as well as bypass secure gateways for internal resources. This is especially helpful, for example, when you have clients using corporate tablets, smartphones, or other mobile devices to browse the web.



Setting up network access

You can force traffic through a tunnel on F5 Access.

Note: *Although you disable* ***Allow local subnet access*** *while enabling* ***Force all traffic through tunnel****, the client still permits local subnet traffic to travel outside of the tunnel. This is a limitation of iOS and not of F5 Access.*

1. On the Main tab, click **Access Policy** > **Network Access** > **Network Access List**.
   The Network Access List screen opens.
2. Click the name to select a network access resource on the Resource List.
   The Network Access editing screen opens.
3. To configure the network settings for the network access resource, click **Network Settings** on the menu bar.
4. To optionally force all traffic through the tunnel, next to **Traffic Options**, enable **Force all traffic through tunnel**.

   If you enable **Use split tunneling for traffic**, you must also specify either a DNS suffix or DNS Address Space pattern to use the VPN DNS servers. If the "DNS Suffix" and "DNS Address Space" fields are

both left blank, then F5 Access does not use the VPN DNS servers and sends all DNS traffic to public DNS servers.

If you enable **Force all traffic through tunnel**, the client uses the proxy server whether you specify the destination using an IP address or a host name.

5. To allow local subnet traffic to bypass the tunnel, select the **Enable** check box for **Allow Local Subnet**. This traffic bypasses the tunnel.
6. For **Client Options**, enable **Client for Microsoft Networks**.
7. Click **Update**.

Prerequisites for configuring F5 Access

Before configuring F5 Access for iOS devices, you must complete the following requirements:

- Set up BIG-IP® Access Policy Manager®.
- Create a network access resource.
- Run the Network Access Setup Wizard.
- Create a connectivity profile.

Additional information about network access and connectivity profiles can be found in the *BIG-IP® Access Policy Manager®: Network Access Configuration* guide.

Access Policy Manager configuration for F5 Access for iOS devices

To configure F5 Access for mobile devices support on BIG-IP® Access Policy Manager®, use the following configuration steps:

- Run the Network Access Setup Wizard.
- Optionally, set up `SSO` and `ACLs` for your network access. Refer to the *BIG-IP® Access Policy Manager® Configuration Guide* on the AskF5™ Knowledge Base for instructions.
- Customize an access policy to support F5 Access.

Running the Network Access Setup wizard

Configure Access Policy Manager® to provide users with full network access from their mobile devices using the Network Access Setup wizard for remote access.

1. On the Main tab, click **Wizards** > **Device Wizards**.
   The Device Wizards screen opens.
2. For Access Policy Manager Configuration, select **Network Access Setup Wizard for Remote Access**, and then click **Next**.
3. In the Basic Properties area of the wizard, clear the **Enable Antivirus Check in Access Policy** check box for Client Side Checks to ensure that your users can connect with F5 Access.
4. Click **Finished**.

You now have network access resource that supports F5 Access for mobile devices.

Customizing an access policy to support F5 Access on Access Policy Manager

Create an access policy that supports F5 Access for iOS.

1. On the Main tab, click **Access Policy** > **Access Profiles**.
   The Access Profiles List screen opens.
2. In the Access Policy column, click the **Edit** link for the access profile you want to configure.
   The visual policy editor opens the access policy in a separate screen.
3. Click the plus **(+)** sign that appears before the `Logon Page` action.
4. Under **Server Side Checks**, select **Client Type**, and click **Add Item**.
5. Click **Save**.
   The Client Type action is added to the access policy, and several new branches appear.
6. On the Edge Client branch of the Client Type action, click the plus **(+)** sign.
7. Under **Server Side Checks**, select **Client OS**, and click **Add Item**.
8. Configure the **iOS** Branch Rule with the configuration objects and resources you want to assign to iOS F5 Access.
9. Click **Finished**, and then click **Save**.
10. Add the network access resource to the branch.
11. Click **Save**.
    This access policy now supports F5 Access for iOS.

Chapter 2

Overview: Access Policies for F5 Access

About access policy branches for F5 Access

You can configure separate access policy branches for F5 Access.

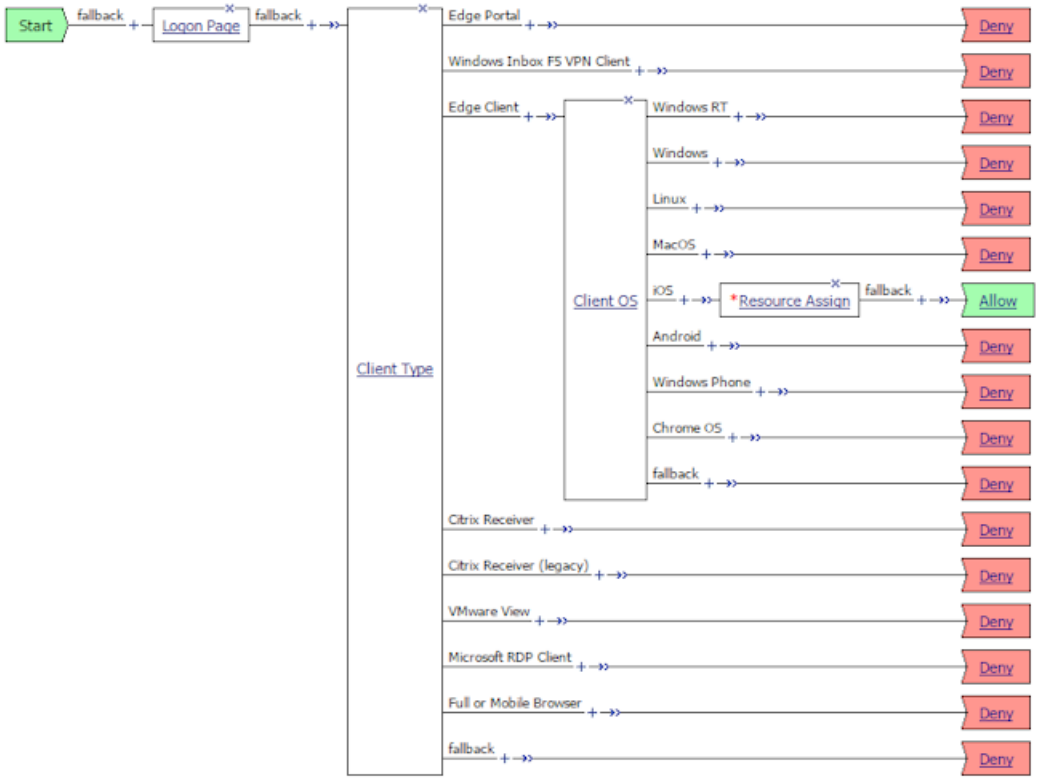
F5 Access for iOS is detected with the following access policy items:

Access policy item	Value
Client Type	F5 Access
Client OS	iOS

Example of basic access policy that supports F5 Access

You can configure an access policy branch to direct iOS device users to F5 Access, and direct non-F5 Access device users to a fallback branch.

This example displays a simple access policy.


Start
fallback
Logon Page
fallback
Client Type
Edge Portal
Deny
Windows Inbox F5 VPN Client
Deny
Edge Client
Client OS
Windows RT
Deny
Windows
Deny
Linux
Deny
MacOS
Deny
iOS
*Resource Assign
fallback
Allow
Android
Deny
Windows Phone
Deny
Chrome OS
Deny
fallback
Deny
Citrix Receiver
Deny
Citrix Receiver (legacy)
Deny
VMware View
Deny
Microsoft RDP Client
Deny
Full or Mobile Browser
Deny
fallback
Deny

Chapter 3

Configuring Per-App VPN with APM and F5 Access

What is per-app VPN?

Apple's VPN framework supports application-level layer-4 tunneling. Apps that are managed by a Mobile Device Manager (MDM) can now be configured to automatically connect to a VPN when they are started. Mobile Safari can be managed for per-app VPN with a configuration profile and without an MDM. Per-app VPN gives IT granular control over corporate network access, and ensures that data transmitted by managed apps travels only through a VPN. Meanwhile, other data, like an employee's personal web browsing activity, does not use the VPN. Per-app VPN also works with Safari on a per-URL basis.

A per-app VPN configuration requires four configuration components.

- A device under MDM management.
- A managed app installed on the device, or Mobile Safari.
- F5 Access for iOS installed on the managed device.
- A per-app VPN profile, and a related F5 Access configuration (VPN). This is configured with an MDM command that associates the app with an F5 Access configuration.

Important: *The managed app and the MDM profile must be deployed with an MDM solution, except in the case of Mobile Safari. The F5 Access configurations may or may not be deployed with an MDM solution. Any app other than Mobile Safari must be installed by the MDM solution, and associated with a VPN configuration.*

About deploying MDM apps over VPNs

The per-app VPN framework allows the administrator to limit VPN access to explicit apps only. Specifically, it allows applications to use one F5 Access configuration (or VPN connection).

Important: *If the F5 Access configuration is not connected when the app starts, all traffic from the app is blocked.*

In practice, some applications may be associated with one F5 Access configuration, and other applications may be associated with other F5 Access configurations.

Important: *Once an app is associated with an F5 Access configuration by the MDM, it must use that VPN only.*

In this example, App 1 or App 2 can be active at the same time, because they use different VPN configurations.

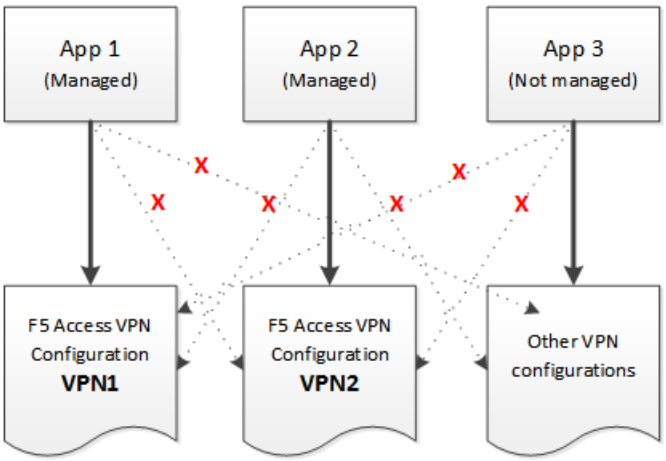


Figure 2: Apps associated with different VPN configurations

Note: *On iOS, you can only activate one L3 VPN cofiguration at a time. However, multiple per-app VPNs can be active and connected simultaneously.*

About access policies for per-app VPN

For per-app VPN, an access policy requires a specific configuration. In particular, the per-app VPN process cannot allow prompts or request information during logon. Therefore, the access policy must be configured to log the user on to the connection without any user interaction.

Creating an access profile

You create an access profile to provide the secured connection between the per-app VPN and the virtual server.

1. On the Main tab, click **Access Policy** > **Access Profiles**.
   The Access Profiles List screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. In the **Name** field, type a name for the access profile.
4. In the Language Settings area, add and remove accepted languages, and set the default language.

   A browser uses the highest priority accepted language. If no browser language matches the accepted languages list, the browser uses the default language.
5. Click **Finished**.

The access profile appears in the Access Profiles List.

Configure the access policy to include a client certificate check.

Adding a client certificate check to the access policy

A client certificate check allows you to authenticate the device to the access policy, without requiring any user interaction that would cause the creation of the per-app VPN tunnel to fail.

1. On the Main tab, click **Access Policy** > **Access Profiles**.
   The Access Profiles List screen opens.
2. In the Access Policy column, click the **Edit** link for the access profile you want to configure to launch the visual policy editor.
   The visual policy editor opens the access policy in a separate screen.
3. Click the **(+)** sign anywhere in the access policy to add a new action item.
   An Add Item screen opens, listing predefined actions on tabs such as General Purpose, Authentication, and so on.
4. Click **Add Item**.
   The screen is not active while the visual policy editor creates the action. The screen closes and a Properties screen displays.
5. Click the **Authentication** tab.
6. Select the **Client Cert Inspection** item, and click **Add Item**.
7. Click **Apply Access Policy** to save your configuration.
8. The properties screen opens. Click **Save**.
9. On the **Successful** branch following the Client Cert Inspection item, click the Deny ending.
10. Change the Deny ending to Allow, and click **Save**.
11. Click **Apply Access Policy** to save your configuration.

The access profile appears in the Access Profiles List.

Configure the virtual server to include this access policy, and make sure the Client SSL profile is enabled on the server.

About setting up Access Policy Manager for per-app VPN

You configure specific settings in the Access Policy Manager® to provide per-app VPN tunnels. Per-app VPN tunnels are SOCKS 5 proxy tunnels, and do not require Network Access resources in the Access Policy. Configure these items on the Access Policy Manager.

- The virtual server must be configured with several critical settings, including an access profile and the **Application Tunnels (Java & Per-App VPN)** setting.
- If there is routing required behind the BIG-IP® device, the SNAT Automap should be enabled.
- The SOCKS proxy has too much access by default, and you should configure ACLs to limit access to only the required URLs and ports.
- You must specify the Client SSL profile on the virtual server. You must also include the same CA bundle on the server that is used to generate the certificate for the client devices.

Configuring a virtual server for per-app VPN

You must have Access Policy Manager® licensed and provisioned.

A virtual server profile enables support for the SOCKS proxy method used by per-app VPN tunnels.

A virtual server profile enables support for the network access used by per-app VPN tunnels.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Click the name of the virtual server you want to modify.
3. 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.
4. From the **Source Address Translation** list, select **Auto Map**.
   The BIG-IP® system uses all of the self IP addresses as the translation addresses for the pool.
5. If you want to provide connections to Java RDP clients for application access, allow Java rewriting for portal access, or support a per-app VPN connection that is configured on a mobile device, select the **Application Tunnels (Java & Per-App VPN)** check box.

   You must enable this setting to make socket connections from a patched Java applet. If your applet doesn't require socket connections, or only uses HTTP to request resources, this setting is not required.
6. In the Access Policy area, from the **Access Profile** list, select the access profile.
7. From the **Connectivity Profile** list, select the connectivity profile.
8. Click **Update** to save the changes.

The virtual server is configured for per-app VPN.

About managing devices

With an MDM, you manage devices by enrolling them. Refer to your MDM documentation to enroll devices.

Important: *A user must enroll the device with the MDM in order for you to manage the device. However, you can deploy VPN configurations to the devices that aren't under management. F5 Access must be installed on the device to deploy configurations or manage the device. F5 Access can be installed either by the user, or deployed with the MDM solution.*

Creating a configuration profile for the managed device

Before you assign a configuration profile to a device, that device must be enrolled with your MDM. Additionally, F5 Access must be installed on the device.

A configuration profile enables the per-app VPN feature on a managed device, and specifies which apps use the VPN.

1. Create a configuration profile for the device.

   Configuration profiles are described at the *Apple Configuration Profile Reference*.
2. In the configuration profile for a particular Per-App VPN configuration, in the VendorConfig section, specify a `PerAppVpn` key with the syntax `<key>PerAppVpn</key><string>true</string>`.

3. Specify an app by sending the `InstallApplication` command or the `Settings` command.

   These settings can be configured only for apps that are installed and managed by the MDM.

4. Specify which managed apps use per-app VPN by sending the `InstallApplication` or `Settings` command.

   Per-app VPN can be specified only for MDM-managed apps. The only exception is Mobile Safari. For Mobile Safari,the admin can specify domains in the profile that start the per-app VPN connection.

5. Specify whether to use Managed User mode, and any settings for Managed User mode, by sending the `ManagedUserConfigurationMode` command, and specifying a custom message. This message can also be localized.
6. Specify a connection screen message, if required, by sending the `ShowConnectionScreenMessage`. This message can also be localized.

Configure Access Policy Manager ®to provide the necessary support for per-app VPN features.

Device identification configuration profile settings

These are settings for identifying devices in an MDM profile.

Device identification settings

Hardware manufacturers have phased out support for many methods of device identification, including UDID, wireless MAC, and others. To identify devices, you can use the device IDs assigned by the MDM.

Table 1: Device identification commands

Key	Type	Description
`MdmAssignedId`	String	The internal device ID assigned to the device by the MDM.
`MdmInstanceId`	String	An arbitrary string that identifies particular MDM instance.
`MdmDeviceUniqueId`	String	The UDID of the iOS device.
`MdmDeviceWifiMacAddress`	String	The wireless MAC address of the device.

Device ID example for iOS

In this example, the commands are deployed in the VendorConfig document.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
...
    <key>VendorConfig</key>
    <dict>
        <key>PerAppVpn</key>
       <string>true</string>
       <key>MdmAssignedId</key>
       <string>MDM assigned ID here</string>
       <key>MdmInstanceId</key>
       <string>some MDM instance ID here</string>
       <key>MdmDeviceUniqueId</key>
       <string>device iOS UDID here</string>
```

```
        <key>MdmDeviceWifiMacAddress</key>
        <string>device wifi mac address here</string>
        <key>MdmDeviceSerialNumber</key>
        <string>device serial number here</string>
    </dict>
...
```

Web logon setting

This setting configures Web Logon mode in an MDM profile.

Web Logon configuration

In the MDM configuration profile, you can use the command WebLogon to specify whether Web Logon is enabled. Use the syntax `<key>WebLogon</key><string>true|false</string>`.

Note: *Web Logon is not supported with Per-App VPN.*

Managed User Configuration Mode configuration profile settings

Settings for Managed User Configuration Mode in an MDM profile.

Managed User Configuration Mode settings

Managed User Configuration Mode restricts users from modifying VPN configurations. When Managed User Configuration Mode is enabled on a device:

- The user cannot add new configurations
- The user cannot delete existing configurations
- The user cannot edit existing configurations
- The URL scheme `create` command cannot be used
- In Managed User Configuration Mode, the user cannot override fields with the URL scheme `start` command, except for the `username` and `password` fields.

Managed User Configuration Mode is enforced through an MDM command. Once Managed User Configuration Mode is enabled by the administrator, all user configurations on the device are removed. Configurations that were pushed to the device by the administrator prior to Managed User Configuration being enabled are preserved. Any Per-App VPN configurations are also preserved.

When Managed User Configuration Mode is enabled, a banner that can be customized by the administrator is displayed on the home screen of the app.

Table 2: Managed User Configuration Mode MDM configuration commands

Key	Type	Description
`ManagedUserConfigurationMode`	[True|False]	Enabled or disable Managed User mode
`ManagedUserConfigurationModeMessage_en`	String	The message in English
`ManagedUserConfigurationModeMessage_es`	String	The message in Spanish
`ManagedUserConfigurationModeMessage_de`	String	The message in German
`ManagedUserConfigurationModeMessage_fr`	String	The message in French
`ManagedUserConfigurationModeMessage_ja`	String	The message in Japanese
`ManagedUserConfigurationModeMessage_ko`	String	The message in Korean

Key	Type	Description
ManagedUserConfigurationModeMessage_zh_r_ch	String	The message in Simplified Chinese
ManagedUserConfigurationModeMessage_zh_r_tw	String	The message in Traditional Chinese

Managed User Configuration Mode examples

In this example, the administrator enables Managed User Configuration Mode on the Python iOS MDM server. Add the following lines to the `setup_commands` function in the `server.py` in the `server` folder.

```
ret_list['Settings - Application Configuration (F5 Access Enable Managed User
 Configuration Mode)'] = dict(
    Command = dict(
        RequestType = 'Settings',
        Settings = [
            dict(
                Item = 'ApplicationConfiguration',
                Identifier = 'com.f5.Edge-Client',
                Configuration = dict(
                    ManagedUserConfigurationMode = 'True',
                    ManagedUserConfigurationModeMessage_en = 'Please do not
                    delete or change configurations in this app. Removal
                    and changes can cause revocation of remote access to
                    your intranet apps.',
                    ManagedUserConfigurationModeMessage_fr = 'Ne supprimez
                    pas ou modifiez les configurations dans cette
                    application. L'élimination et les modifications peuvent
                    provoquer la révocation de l'accès à distance à vos
                    applications intranet.',
                )
            )
        ]
    )
)
```

In this example, the administrator disables Managed User Configuration Mode on the Python iOS MDM server. Add the following lines to the `setup_commands` function in the `server.py` in the `server` folder.

```
ret_list['Settings - Application Configuration (F5 Access Disable Managed User
 Configuration Mode)'] = dict(
    Command = dict(
        RequestType = 'Settings',
        Settings = [
            dict(
                Item = 'ApplicationConfiguration',
                Identifier = 'com.f5.Edge-Client',
                Configuration = dict(
                    ManagedUserConfigurationMode = 'False',
                )
            ),
        ]
    )
)
```

Connection screen message configuration profile settings

Settings for the connection message in an MDM profile.

Connection screen message settings

An optional message can be displayed on the connection screen for informational purposes. This feature is enabled by an MDM command, and configured through the MDM profile.

The message is displayed above the Enterprise VPN section on the connection screen screen. The interface is scrollable so the controls can still be accessed, including Per-App VPN items. The message is limited to nine lines of text.

Table 3: Connection screen message commands

Key	Type	Description
`ShowConnectionScreenMessage`	[True|False]	Enables or disables the message on the connection screen
`ConnectionScreenMessage_en`	String	The message displayed on the connection screen in English
`ConnectionScreenMessage_es`	String	The message displayed on the connection screen in Spanish
`ConnectionScreenMessage_de`	String	The message displayed on the connection screen in German
`ConnectionScreenMessage_fr`	String	The message displayed on the connection screen in French
`ConnectionScreenMessage_ja`	String	The message displayed on the connection screen in Japanese
`ConnectionScreenMessage_ko`	String	The message displayed on the connection screen in Korean
`ConnectionScreenMessage_zh_r_ch`	String	The message displayed on the connection screen in Simplified Chinese
`ConnectionScreenMessage_zh_r_tw`	String	The message displayed on the connection screen in Traditional Chinese

Connection Screen Message examples

In this example, the administrator shows a localized Connection Screen Message on the Python iOS MDM server. Add the following lines to the `setup_commands` function in the `server.py` in the `server` folder.

```
ret_list['Settings - Application Configuration (F5 Access Enable Connection
Screen Message)'] = dict(
    Command = dict(
        RequestType = 'Settings',
        Settings = [
            dict(
                Item = 'ApplicationConfiguration',
                Identifier = 'com.f5.Edge-Client',
                Configuration = dict(
                    ShowConnectionScreenMessage = 'True',
                    ConnectionScreenMessage_en = 'Please do not delete
                    or change configurations in this app. Removal and
                    changes can cause revocation of remote access to
                    your intranet apps.',
                    ConnectionScreenMessage_fr = 'Ne supprimez pas
                    ou modifiez les configurations dans cette
                    application. L'élimination et les modifications peuvent
```

```
                    provoquer la révocation de l'accès à distance à vos
                    applications intranet.',
                )
            )
        ]
    )
)
```

In this example, the administrator disables the Connection Screen Message on the Python iOS MDM server. Add the following lines to the `setup_commands` function in the `server.py` in the `server` folder.

```
ret_list['Settings - Application Configuration (F5 Access Disable Connection
 Screen Message)'] = dict(
    Command = dict(
        RequestType = 'Settings',
        Settings = [
            dict(
                Item = 'ApplicationConfiguration',
                Identifier = 'com.f5.Edge-Client',
                Configuration = dict(
                    ShowConnectionScreenMessage = 'False',
                )
            ),
        ]
    )
)
```

Per-App VPN configuration profile settings

Settings for the per-app VPN profile in an MDM.

Per-App VPN settings

The per-app VPN payload supports all of the keys described in the *Apple Configuration Profile Reference*. These keys, specific to the per-app VPN payload, are described in that reference as well.

Table 4: Per-App VPN specific keys

Key	Type	Description
VPNUUID	String	A globally-unique identifier for this VPN configuration. This identifier is used to configure apps so that they use the per-app VPN service for all of their network communication.
OnDemandMatchAppEnabled	Boolean	If **true**, the per-app VPN connection starts automatically when apps linked to this per-app VPN service initiate network communication. If **false**, the per-app VPN connection must be started manually by the user before apps linked to this per-app VPN service can initiate network communication. If this key is not present, the value of the OnDemandEnabled key is used to determine the status of per-app VPN On Demand.
SafariDomains	Array	This optional key is a special case of App-to-Per App VPN Mapping. It sets up the app mapping for Safari

Key	Type	Description
		with a specific identifier and a designated requirement. The array contains strings, each of which is a domain that triggers a VPN connection in Safari. Do not specify a full URI; rule matching works only with the domain name. The rule matching behavior is as follows: • Before being matched against a host, all leading and trailing dots are stripped from the domain string. For example, if the domain string is "`.com`" the domain string used to match is "`com`". • Each label in the domain string must match an entire label in the host string. For example, a domain of "`example.com`" matches "`www.example.com`", but not "`old.badexample.com`". • Domain strings with only one label must match the entire host string. For example, a domain of "`com`" matches "`com`", not "`www.example.com`".

Example per-app VPN configuration profile

Includes a sample configuration profile for the per-app VPN configuration profile.

Per-App VPN configuration example profile

The following example uses sample data only. For your own configuration, items like the `PayloadDisplayName`, `Payload UUID`, `UserDefinedName`, and the user name, password and certificate information must be customized to your network and installation.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>PayloadContent</key>
    <array>
      <dict>
        <key>IPv4</key>
        <dict>
          <key>OverridePrimary</key>
          <integer>0</integer>
        </dict>
        <key>PayloadDescription</key>
        <string>Configures VPN settings, including authentication.</string>
        <key>PayloadDisplayName</key>
        <string>VPN (Per-App VPN Test)</string>
        <key>PayloadIdentifier</key>
        <string>com.example.mdm.perapp.vpn.vpn</string>
        <key>PayloadOrganization</key>
        <string></string>
        <key>PayloadType</key>
        <string>com.apple.vpn.managed.applayer</string>
        <key>PayloadUUID</key>
        <string>5A015006-D559-4C5C-B197-737CF4DCFA96</string>
        <key>PayloadVersion</key>
        <integer>1</integer>
```

```
        <key>Proxies</key>
        <dict/>
        <key>UserDefinedName</key>
        <string>Per-App VPN Test</string>
        <key>VPN</key>
        <dict>
          <key>AuthName</key>
          <string>testuser</string>
          <key>AuthPassword</key>
          <string>testpassword</string>
          <key>AuthenticationMethod</key>
          <string>Certificate</string>
          <key>PayloadCertificateUUID</key>
          <string>C9BF4927-E819-4521-88DE-2AEB6E1DC3D8</string>
          <key>RemoteAddress</key>
          <string>vpn.example.com</string>
          <key>OnDemandMatchAppEnabled</key>
          <true/>
      </dict>
      <key>VPNSubType</key>
      <string>com.f5.F5-Edge-Client.vpnplugin</string>
      <key>VPNType</key>
      <string>VPN</string>
      <key>VendorConfig</key>
      <dict>
          <key>PerAppVpn</key>
          <string>true</string>
        </dict>
        <key>SafariDomains</key>
        <array>
          <string>safaridomain1.com</string>
          <string>safaridomain2.com</string>
        </array>
        <key>VPNUUID</key>
        <string>9F658A35-2B0F-4D5E-872D-61A9130FE882</string>
      </dict>
      <dict>
        <key>Password</key>
        <string>123456</string>
        <key>PayloadCertificateFileName</key>
        <string>identity.p12</string>
        <key>PayloadContent</key>
        <data>
            MIIL2QIBAzCCC58GCSqGSIb3DQEHAaCCC5AEgguMMIILiDCCBj8G
            .............<truncated for example>.............
            hxd6YPi7JKB/24dSls9gKO/DHVoECHap2RUyKvQTAgIIAA==
            </data>
        <key>PayloadDescription</key>
        <string>Provides device authentication (certificate or
identity).</string>
        <key>PayloadDisplayName</key>
        <string>identity.p12</string>
        <key>PayloadIdentifier</key>
        <string>com.f5.mdm.perapp.vpn.credential</string>
        <key>PayloadOrganization</key>
        <string/>
        <key>PayloadType</key>
        <string>com.apple.security.pkcs12</string>
        <key>PayloadUUID</key>
        <string>C9BF4927-E819-4521-88DE-2AEB6E1DC3D8</string>
        <key>PayloadVersion</key>
        <integer>1</integer>
      </dict>
    </array>
    <key>PayloadDescription</key>
    <string>PerApp VPN Payload Test</string>
    <key>PayloadDisplayName</key>
    <string>MDM - Per-App VPN</string>
    <key>PayloadIdentifier</key>
```

```
    <string>com.f5.mdm.perapp.vpn</string>
    <key>PayloadOrganization</key>
    <string/>
    <key>PayloadRemovalDisallowed</key>
    <false/>
    <key>PayloadType</key>
    <string>Configuration</string>
    <key>PayloadUUID</key>
    <string>06A850CC-BC81-43FB-AA16-42BE472D2421</string>
    <key>PayloadVersion</key>
    <integer>1</integer>
  </dict>
</plist>
```

Chapter 4

Additional Access Policy Manager Configuration Information

F5 Access for iOS session variables

The following table contains a list of session variables and their attributes.

Session variable	Description
session.client.type	Indicates the client type, for example `Standalone`.
session.client.platform	Indicates the platform type, such as `iOS`.
session.user.agent	Indicates the browser, device type, and operating system version of the client, as well as the version of F5 Access.
session.client.model	Indicates the model number of the mobile device. For example, `iPhone`
session.client.platform_version	Indicates the platform and version of the mobile device. For example, 10.2.1
session.client.unique_id	Indicates the unique ID of the device. For example, RC1KQLCJFOJEEM0XIOB3P52OMUQ3UN9Y3SDA5RWR.
session.client.jailbreak	Indicates the jailbreak status of the device. `0` indicates the device is not jailbroken, `1` indicates the device is jailbroken, and an empty response indicates that the status of the device is unknown.
session.client.biometric_fingerprint	Indicates whether the device supports biometric fingerprint authentication. `1` indicates that a fingerprint is configured, `0` indicates that a fingerprint is not configured, or the device does not support fingerprint authentication.
session.client.vpn_scope	Indicates the scope of the VPN tunnel. The result is `device` for a device-wide VPN connection, and `per-app` for a per-app VPN.
session.client.vpn_tunnel_type	Indicates the type of VPN tunnel. For F5 Access for iOS, this can be `L3` for a standard connection, or `L4` for a Per-App VPN connection.
session.client.vpn_start_type	Indicates how the VPN connection was initiated. • `manual` - Indicates that the connection was initiated by the user.

Session variable	Description
	• `on-demand` - Indicates that the connection was initiated by Per-App VPN or an MDM.
session.client.device_passcode_set	Indicates whether the user has a device unlock passcode, PIN, or biometric authentication configured. The results is `1` if a device lock is configures, and `0` if it is not.

Access Policy Manager configuration tips

The following table provides tips for setting up F5 Access for devices.

Feature	Information
VPN On-Demand (iOS only)	A connection cannot be established if the server has an invalid certificate. To work around this issue, manually import the invalid certificate onto the device.
Client endpoint checks	Client end-point checks are not currently supported.
Require Device Authentication	For devices with iOS 9 or later, F5 Access can require device authentication with one of the device locking methods, including biometric authentication (Touch ID), a PIN, or a passphrase. To enable device authentication for F5 Access, in the **Connectivity Profile** under **iOS Edge Client**, enable the options **Allow Password Caching** and **Require Device Authentication**.

Password caching policy	• Under **Client Policy**, if **Enforce session settings** is not enabled, clients can save their encrypted password to disk, regardless of what settings are configured under **Session Settings**.
	• Under `Password Caching Options`, if you set **Cache password within application for** for a specific amount of time, after a successful logon the submitted credentials are cached until one of the following events occurs:
	  • The specified credential cache duration expires.
	  • The server address of the configuration within the application changes.
	  • The user name of the configuration within the application changes.
	  • The F5 Access user switches between configurations and makes a new connection.
	  • The configuration is deleted and a new one is created.
	• On the device, even if a user clicks **Disconnect**, then terminates the application or restarts the device, cached credentials are not cleared until the specified cache time expires.

Client certificates	Client certificate authentication is supported, either with a certificate alone or with a certificate secured with a user name and password.
On-Demand Cert Auth	If used, the `On-Demand Cert Auth` action must be placed after other authentication actions in the access policy.

About starting the client from a URL scheme

You can start F5 Access connections for users from a URL. You can then provide these URLs to users, so they can start the VPN connection without having to manually start the application. If there is already an active connection, a prompt appears to warn the user that the existing connection must be stopped before the new connection can start. The connection uses a client certificate if it is specified in the existing configuration.

Note: *For F5 Access clients, you can use the URL schemes* `f5access://` *and* `f5edgeclient://`*. Both provide the same functionality.*

URL connections use the following parameters. This is an example, you must provide your own parameters and values.

```
f5access://{start|stop}?[parameter1=value1&parameter2=value2...]
```

Note: *Special characters in parameters must be URL-encoded.*

You can start an alternate light client with no client branding, using the following parameters.

```
f5access-lite://{start|stop}?[parameter1=value1&parameter2=value2...]
```

Note: *Special characters in parameters must be URL-encoded.*

The syntax to start a connection from a URL follows.

`start`
Starts a connection. The `start` command requires either the `name` or `server` parameter to be present in the URL. If the `name` parameter is specified, then F5 Access looks for the name in the list of existing configuration entries. If the `server` parameter is specified, then the `name` parameter is set to the same value as the `server` parameter. A new configuration is created if a configuration with that name does not exist. If the specified configuration already exists, the other parameters specified in the URL are merged with the existing configuration. The result of this merged configuration is used only for the current, active connection, and does not persist. If a `name` is specified with other parameters, such as `server`, `username`, or `password`, those parameters override what is specified in the configuration.

`sid`
A parameter used to specify the session ID with which to start the connection. When the parameter `sid` is provided, the `username` and `password` parameters are ignored, and no additional authentication occurs.

`username`
A parameter used to specify the user name with which to start the connection. When the `username` is specified without a `password`, then an authentication prompt is displayed.

`password`
A parameter used to specify the password with which to start the connection. When the `password` parameter is specified, it is used as a one-time password and not saved in the configuration.

`postlaunch_url`
A parameter used to specify the URL that starts after the connection starts.

logon_mode
An optional parameter that specifies whether the logon mode is the standard logon (native) or web logon (web). The default logon mode is native.

Examples of starting a client from a URL

The following examples illustrate how to start F5 Access and F5 Access Lite connections for users from a URL.

Connecting to an existing configuration called MYVPN:

```
f5access://start?name=MYVPN
```

Connecting to an existing configuration called MYVPN with F5 Access Lite:

```
f5access-lite://start?name=MYVPN
```

Connecting to an existing configuration called MYVPN with F5 Access Lite, from an app registered with the URL scheme refapp://, and specifying x-callback-url parameters:

```
f5access-lite://start?name=MYVPN&x-success=refapp%3A%2F%2Fsuccess
&x-cancel=refapp%3A%2F%2Fcancel&x-error=refapp%3A%2F%2Ferror
```

Connecting to an existing configuration called MYVPN and including the server URL myvpn.siterequest.com:

```
f5access://start?name=MYVPN&server=myvpn.siterequest.com
```

Connecting to a specific server called myvpn.siterequest.com:

```
f5access://start?server=myvpn.siterequest.com
```

Connecting to a specific server called myvpn.siterequest.com with web logon enabled:

```
f5access://start?server=myvpn.siterequest.com&logon_mode=web
```

Connecting to an existing configuration called MYVPN and including the username smith and the password passw0rd:

```
f5access://start?name=MYVPN&username=smith&password=passw0rd
```

Starting a connection to a configuration called MYVPN and specifying the post-launch URL jump://?host=10.10.1.10&username=smith:

```
f5access://start?name=MYVPN&postlaunch_url=jump%3A%2F%2F%3Fhost%3D10.10.1.10
%26username%3Dsmith
```

Stopping a connection:

```
f5access://stop
```

About F5 Access Lite mode

You can use a URL parameter to start F5 Access Lite mode. F5 Access Lite removes branding and presents a plain black screen for F5 Access. In F5 Access Lite mode, the client has certain features and restrictions.

f5access-lite://
Provide a URL parameter that begins with `f5access-lite://` to start a connection in F5 Access Lite mode.

Plain black screen
When the app starts from an `f5access-lite://` URL, a black screen appears for the F5 Access configuration, hiding most controls from the user.

Text displayed
The F5 Access Lite screen displays status messages while the connection starts, and a **Cancel** button.

Authentication
Authentication or confirmation prompts appear over the black screen.

Standard F5 Access interface available after app switch
If the user switches away from the F5 Access, and then returns to it, the standard F5 Access interface shows, not blacked out. The F5 Access Lite user interface displays at initial logon only.

x-callback-url
The `x-callback` URL allows you to send one or more messages back to another app that has a registered URL scheme. The available messages are `x-success`, `x-cancel`, and `x-error`.

Standard logon mode only
F5 Access Lite does not work with web logon mode.

About defining a server from a URL

You can add BIG-IP® server definitions to F5 Access from a URL. You can provide these URLs to users, so they can create and/or start VPN connections without having to manually start the application.

Use the following URL and parameters to create a server:

Note: *Special characters in parameters must be URL-encoded.*

You can start an alternate light client with no client branding, using the following parameters.

```
f5access-lite://create?server=server_address[&parameter1=value1&parameter2=value2...]
```

Note: *Special characters in parameters must be URL-encoded.*

The syntax to define a server from a URL follows.

`server`
The server address is either a DNS name or an IP address.

`name`
An optional description of the server.

username
An optional parameter used to specify the user name with which to start the connection. When the `username` is specified without a `password`, then an authentication prompt is displayed. If no `username` is specified during server creation, the user is prompted for it at session initiation, if required.

password
An optional parameter used to specify the password with which to start the server connection. When the `password` parameter is specified, it is used as a one-time password and not saved in the configuration.

certcn

Certificate common name. Matches the common name of the issuer of a valid certificate pre-installed on the device.

Important: *Only one of* `certcn`, `cert_url`, *or* `cert_keychain_alias` *can be specified.*

---

logon_mode

Specifies whether the logon mode is the standard logon (`native`) or web logon (`web`). The default logon mode is `native`.

Examples of defining a server from a URL

The following examples illustrate how to define servers for F5 Access connections from a URL.

Create a server at `edgeportal.siterequest.com`:

```
f5access://create?server=edgeportal.siterequest.com
```

Create a server named `EdgePortal` with the server URL `edgeportal.siterequest.com`:

```
f5access://create?name=EdgePortal&server=edgeportal.siterequest.com
```

Create the same server with a user name, password, and certificate:

```
f5access://create?name=EdgePortal&server=edgeportal.siterequest.com
&username=edgeportal&password=iosdemo&certcn=clientcert-cert.siterequest.com
```

Create the same server with a user name and certificate:

```
f5access://create?name=EdgePortal&server=edgeportal.siterequest.com
&username=edgeportal&certcn=clientcert-cert.siterequest.com
```

Index

S

T

U

V

W