Network Working Group   S. Yadav
Request for Comments: 2752   R. Yavatkar
Category: Standards Track   Intel
                                                           R. Pabbati
                                                             P. Ford
                                                             T. Moore
                                                           Microsoft
                                                            S. Herzog
                                                            IPHighway
                                                        January 2000

                    Identity Representation for RSVP

Status of this Memo

Copyright Notice

Abstract

   This document describes the representation of identity information in
   POLICY_DATA object [POL-EXT] for supporting policy based admission
   control in RSVP. The goal of identity representation is to allow a
   process on a system to securely identify the owner and the
   application of the communicating process (e.g. user id) and convey
   this information in RSVP messages (PATH or RESV) in a secure manner.
   We describe the encoding of identities as RSVP policy element. We
   describe the processing rules to generate identity policy elements
   for multicast merged flows. Subsequently, we describe representations
   of user identities for Kerberos and Public Key based user
   authentication mechanisms. In summary we describe the use of this
   identity information in an operational setting.

1. Conventions used in this document

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
   document are to be interpreted as described in [RFC-2119].

2. Introduction

   RSVP [RFC 2205] is a resource reservation setup protocol designed for
   an integrated services Internet [RFC 1633]. RSVP is used by a host to
   request specific quality of service (QoS) from the network for
   particular application data streams or flows. RSVP is also used by
   routers to deliver QoS requests to all nodes along the path(s) of the
   flows and to establish and maintain state to provide the requested
   service. RSVP requests will generally result in resources being
   reserved in each node along the data path. RSVP allows particular
   users to obtain preferential access to network resources, under the
   control of an admission control mechanism. Permission to make a
   reservation is based both upon the availability of the requested
   resources along the path of the data and upon satisfaction of policy
   rules. Providing policy based admission control mechanism based on
   user identity or application is one of the prime requirements.

   In order to solve these problems and implement identity based policy
   control it is required to identify the user and/or application making
   a RSVP request.

   This document proposes a mechanism for sending identification
   information in the RSVP messages and enables authorization decisions
   based on policy and identity.

   We describe the authentication policy element (AUTH_DATA) contained
   in the POLICY_DATA object. User process can generate an AUTH_DATA
   policy element and gives it to RSVP process (service) on the
   originating host. RSVP service inserts AUTH_DATA into the RSVP
   message to identify the owner (user and/or application) making the
   request for network resources. Network elements, such as routers,
   authenticate request using the credentials presented in the AUTH_DATA
   and admit the RSVP message based on admission policy. After a request
   has been authenticated, first hop router installs the RSVP state and
   forwards the new policy element returned by the Policy Decision Point
   (PDP) [POL-FRAME].

3. Policy Element for Authentication Data

3.1 Policy Data Object Format

   POLICY_DATA objects contain policy information and are carried by
   RSVP messages. A detail description of the format of POLICY_DATA
   object can be found in "RSVP Extensions for Policy Control" [POL-
   EXT].

3.2 Authentication Data Policy Element

   In this section, we describe a policy element (PE) called
   authentication data (AUTH_DATA). AUTH_DATA policy element contains a
   list of authentication attributes.

```
        +-------------+-------------+-------------+-------------+
        | Length                    | P-Type = Identity Type    |
        +-------------+-------------+-------------+-------------+
        // Authentication Attribute List                       //
        +-------------------------------------------------------+
```

   Length
        The length of the policy element (including the Length and P-
        Type) is in number of octets (MUST be a multiple of 4) and
        indicates the end of the authentication attribute list.

   P-Type (Identity Type)
        Type of identity information contained in this Policy Element
        supplied as the Policy element type (P-type). The Internet
        Assigned Numbers Authority (IANA) acts as a registry for policy
        element types for identity as described in the [POL-EXT].
        Initially, the registry contains the following P-Types for
        identity:

        1   AUTH_USER   Authentication scheme to identify users

        2   AUTH_APP   Authentication scheme to identify
                             applications

   Authentication Attribute List

        Authentication attributes contain information specific to
        authentication method and type of AUTH_DATA. The policy element
        provides the mechanism for grouping a collection of
        authentication attributes.

3.3 Authentication Attributes

   Authentication attributes MUST be encoded as a multiple of 4 octets,
   attributes that are not a multiple of 4 octets long MUST be padded to
   a 4-octet boundary.

```
        +--------+--------+--------+--------+
        | Length          | A-Type |SubType |
        +--------+--------+--------+--------+
        | Value ...
        +--------+--------+--------+--------+
```

Length
     The length field is two octets and indicates the actual length of
     the attribute (including the Length and A-Type fields) in number
     of octets. The length does not include any bytes padding to the
     value field to make the attribute multiple of 4 octets long.

A-Type
     Authentication attribute type (A-Type) field is one octet. IANA
     acts as a registry for A-Types as described in the section 9,
     IANA Considerations. Initially, the registry contains the
     following A-Types:

          1   POLICY_LOCATOR   Unique string for locating the
                                  admission policy (such as X.500 DN
                                  described in [RFC 1779]).

          2   CREDENTIAL   User credential such as Kerberos
                                  ticket, or digital certificate.
                                  Application credential such as
                                  application ID.

          3   DIGITAL_SIGNATURE   Digital signature of the
                                  authentication data policy element.

          4   POLICY_ERROR_OBJECT Detailed information on policy
                                  failures.

SubType
     Authentication attribute sub-type field is one octet. Value of
     SubType depends on A-type.

Value:
     The value field contains the attribute specific information.

3.3.1 Policy Locator

   POLICY_LOCATOR is used to locate the admission policy for the user
   or application. Distinguished Name (DN) is unique for each User or
   application hence a DN is used as policy locator.

```
   +-------+-------+-------+-------+
   | Length        |A-Type |SubType|
   +-------+-------+-------+-------+
   | OctetString ...
   +-------+-------+-------+--------
```

Length
     Length of the attribute, which MUST be >= 4.

A-Type
     POLICY_LOCATOR

SubType
     Following sub types for POLICY_LOCATOR are defined. IANA acts as
     a registry for POLICY_LOCATOR sub types as described in the
     section 9, IANA Considerations. Initially, the registry contains
     the following sub types for POLICY_LOCATOR:

     1 ASCII_DN   OctetString contains the X.500 DN as described
                        in the RFC 1779 as an ASCII string.

     2 UNICODE_DN   OctetString contains the X.500 DN described in
                        the RFC 1779 as an UNICODE string.

     3 ASCII_DN_ENCRYPT OctetString contains the encrypted X.500
                        DN. The Kerberos session key or digital
                        certificate private key is used for encryption.
                        For Kerberos encryption the format is the same
                        as returned from gss_seal [RFC 1509].

     4 UNICODE_DN_ENCRYPT OctetString contains the encrypted
                        UNICODE X.500 DN. The Kerberos session key or
                        digital certificate private key is used for
                        encryption. For Kerberos encryption the format
                        is the same as returned from gss_seal [RFC
                        1509].

   OctetString
     The OctetString field contains the DN.

3.3.2 Credential

   CREDENTIAL indicates the credential of the user or application to be
   authenticated. For Kerberos authentication method the CREDENTIAL
   object contains the Kerberos session ticket. For public key based
   authentication this field contains a digital certificate.

   A summary of the CREDENTIAL attribute format is shown below. The
   fields are transmitted from left to right.

```
+-------+-------+-------+-------+
| Length         |A-Type |SubType|
+-------+-------+-------+-------+
| OctetString ...
+-------+-------+-------+--------
```

Length
    Length of the attribute, which MUST be >= 4.

A-Type
    CREDENTIAL

SubType
    IANA acts as a registry for CREDENTIAL sub types as described in
    the section 9, IANA Considerations. Initially, the registry
    contains the following sub types for CREDENTIAL:

    1 ASCII_ID   OctetString contains user or application
                       identification in plain ASCII text string.

    2 UNICODE_ID   OctetString contains user or application
                       identification in plain UNICODE text string.

    3 KERBEROS_TKT OctetString contains Kerberos ticket.

    4 X509_V3_CERT OctetString contains X.509 V3 digital
                       certificate [X.509].

    5 PGP_CERT   OctetString contains PGP digital certificate.

    OctetString
    The OctetString contains the user or application credential.

3.3.3 Digital Signature

   The DIGITAL_SIGNATURE attribute MUST be the last attribute in the
   attribute list and contains the digital signature of the AUTH_DATA
   policy element. The digital signature signs all data in the
   AUTH_DATA policy element up to the DIGITAL_SIGNATURE. The algorithm
   used to compute the digital signature depends on the authentication
   method specified by the CREDENTIAL SubType field.

   A summary of DIGITAL_SIGNATURE attribute format is described below.

```
+-------+-------+-------+-------+
| Length         |A-Type |SubType|
+-------+-------+-------+-------+
| OctetString ...
+-------+-------+-------+--------
```

Length
    Length of the attribute, which MUST be >= 4.

    ti3 A-Type
    DIGITAL_SIGNATURE

SubType
    No sub types for DIGITAL_SIGNATURE are currently defined. This
    field MUST be set to 0.

OctetString
    OctetString contains the digital signature of the AUTH_DATA.

3.3.4 Policy Error Object

    This attribute is used to carry any specific policy control errors
    generated by a node when processing/validating an Authentication Data
    Policy Element. When a RSVP policy node (local policy decision point
    or remote PDP) encounters a request that fails policy control due to
    its Authentication Policy Element, it SHOULD add a POLICY_ERROR_CODE
    containing additional information about the reason the failure
    occurred into the policy element. This will then cause an appropriate
    PATH_ERROR or RESV_ERROR message to be generated with the policy
    element and appropriate RSVP error code in the message, which is
    returned to the request's source.

    The AUTH_DATA policy element in the PATH or RSVP message SHOULD not
    contain the POLICY_ERROR_OBJECT attribute. These are only inserted
    into PATH_ERROR and RESV_ERROR messages when generated by policy
    aware intermediate nodes.

```
+----------+----------+----------+----------+
| Length              | A-Type   |SubType(0)|
+----------+----------+----------+----------+
| 0 (Reserved)        | ErrorValue          |
+----------+----------+----------+----------+
| OctetString ...
+----------+----------+----------+----------+
```

Length
    Length of the attribute, which MUST be >= 8.

A-Type
     POLICY_ERROR_CODE

ErrorValue
     A 32-bit bit code containing the reason that the policy decision
     point failed to process the policy element. Following values have
     been defined.

     1 ERROR_NO_MORE_INFO   No information is available.
     2 UNSUPPORTED_CREDENTIAL_TYPE This type of credentials is
                                     not supported.

     3 INSUFFICIENT_PRIVILEGES   The credentials do not have
                                     sufficient privilege.

     4 EXPIRED_CREDENTIAL   The credential has expired.

     5 IDENTITY_CHANGED   Identity has changed.

OctetString
     The OctetString field contains information from the policy
     decision point that MAY contain additional information about the
     policy failure. For example, it may include a human-readable
     message in the ASCII text.

4. Authentication Data Formats

   Authentication attributes are grouped in a policy element to
   represent the identity credentials.

4.1 Simple User Authentication

   In simple user authentication method the user login ID (in plain
   ASCII or UNICODE text) is encoded as CREDENTIAL attribute. A summary
   of the simple user AUTH_DATA policy element is shown below.

```
+-------------+-------------+-------------+-------------+
| Length                    | P-type = AUTH_USER        |
+-------------+-------------+-------------+-------------+
| Length                    |POLICY_LOCATOR| SubType    |
+-------------+-------------+-------------+-------------+
| OctetString (User's Distinguished Name) ...
+-------------+-------------+-------------+-------------+
| Length                    |CREDENTIAL   | ASCII_ID    |
+-------------+-------------+-------------+-------------+
| OctetString (User's login ID) ...
+-------------+-------------+-------------+-------------+
```

4.2 Kerberos User Authentication

   Kerberos [RFC 1510] authentication uses a trusted third party (the
   Kerberos Distribution Center - KDC) to provide for authentication of
   the user to a network server. It is assumed that a KDC is present and
   both host and verifier of authentication information (router or PDP)
   implement Kerberos authentication.

   A summary of the Kerberos AUTH_DATA policy element is shown below.

```
   +--------------+--------------+--------------+--------------+
   | Length                      | P-type = AUTH_USER          |
   +--------------+--------------+--------------+--------------+
   | Length                      |POLICY_LOCATOR|    SubType   |
   +--------------+--------------+--------------+--------------+
   | OctetString (User's Distinguished Name) ...
   +--------------+--------------+--------------+--------------+
   | Length                      | CREDENTIAL   | KERBEROS_TKT |
   +--------------+--------------+--------------+--------------+
   | OctetString (Kerberos Session Ticket) ...
   +--------------+--------------+--------------+--------------+
```

4.2.1. Operational Setting using Kerberos Identities

   An RSVP enabled host is configured to construct and insert AUTH_DATA
   policy element into RSVP messages that designate use of the Kerberos
   authentication method (KERBEROS_TKT). Upon RSVP session
   initialization, the user application contacts the KDC to obtain a
   Kerberos ticket for the next network node or its PDP. A router when
   generating a RSVP message contacts the KDC to obtain a Kerberos
   ticket for the next hop network node or its PDP. The identity of the
   PDP or next network hop can be statically configured, learned via
   DHCP or maintained in a directory service. The Kerberos ticket is
   sent to the next network node (which may be a router or host) in a
   RSVP message. The KDC is used to validate the ticket and
   authentication the user sending RSVP message.

4.3 Public Key based User Authentication

   In public key based user authentication method digital certificate is
   encoded as user credentials. The digital signature is used for
   authenticating the user. A summary of the public key user AUTH_DATA
   policy element is shown below.

```
+-------------+-------------+-------------+-------------+
| Length                    | P-type = AUTH_USER        |
+-------------+-------------+-------------+-------------+
| Length                    |POLICY_LOCATOR|   SubType  |
+-------------+-------------+-------------+-------------+
| OctetString (User's Distinguished Name) ...
+-------------+-------------+-------------+-------------+
| Length                    | CREDENTIAL  |   SubType   |
+-------------+-------------+-------------+-------------+
| OctetString (User's Digital Certificate) ...
+-------------+-------------+-------------+-------------+
| Length                    |DIGITAL_SIGN. | 0          |
+-------------+-------------+-------------+-------------+
| OctetString (Digital signature) ...
+-------------+-------------+-------------+-------------+
```

4.3.1. Operational Setting for public key based authentication

   Public key based authentication assumes following:

      - RSVP service requestors have a pair of keys (private key and
        public key).

      - Private key is secured with the user.

      - Public keys are stored in digital certificates and a trusted
        party, certificate authority (CA) issues these digital
        certificates.

      - The verifier (PDP or router) has the ability to verify the
        digital certificate.

   RSVP requestor uses its private key to generate DIGITAL_SIGNATURE.
   User Authenticators (router, PDP) use the user's public key (stored
   in the digital certificate) to verify the signature and authenticate
   the user.

4.4 Simple Application Authentication

   The application authentication method encodes the application
   identification such as an executable filename as plain ASCII or
   UNICODE text.

```
+---------------+-------------+-------------+-------------+
| Length                      | P-type = AUTH_APP         |
+---------------+-------------+-------------+-------------+
| Length                      |POLICY_LOCATOR| SubType    |
+---------------+-------------+-------------+-------------+
| OctetString (Application Identity attributes in
|             the form of  a Distinguished Name) ...
+---------------+-------------+-------------+-------------+
| Length                      | CREDENTIAL   | ASCII_ID   |
+---------------+-------------+-------------+-------------+
| OctetString (Application Id, e.g., vic.exe)
+---------------+-------------+-------------+-------------+
```

5. Operation

```
+-----+                                            +-----+
| PDP |-------+                                     | PDP |
+-----+       |          ...................        +-----+
              |          :                 :           |
         +--------+      :    Transit       :      +-------+
     +----| Router |------:    Network       : -------| Router|--+
     |    +--------+      :                 :      +-------+   |
     |         |          :.................:           |      |
     |         |                                        |      |
   Host A      B                                        C      D
```
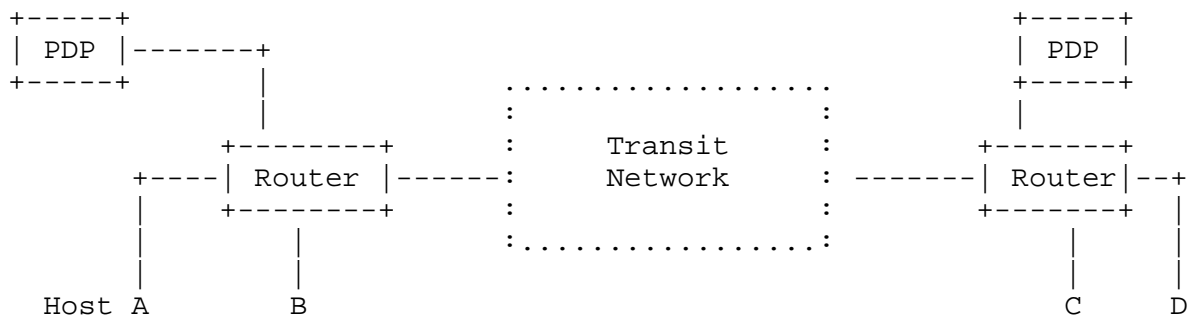
    Figure 1: User and Application Authentication using AUTH_DATA PE

    Network nodes (hosts/routers) generate AUTH_DATA policy elements,
    contents of which are depend on the identity type used and the
    authentication method used. These generally contain authentication
    credentials (Kerberos ticket or digital certificate) and policy
    locators (which can be the X.500 Distinguished Name of the user or
    network node or application names). Network nodes generate AUTH_DATA
    policy element containing the authentication identity when making the
    RSVP request or forwarding a RSVP message.

    Network nodes generate user AUTH_DATA policy element using the
    following rules

    1. For unicast sessions the user policy locator is copied from the
       previous hop. The authentication credentials are for the current
       network node identity.

    2. For multicast messages the user policy locator is for the current
       network node identity. The authentication credentials are for the
       current network node.

Network nodes generate application AUTH_DATA policy element using the following rules:

1. For unicast sessions the application AUTH_DATA is copied from the previous hop.

2. For multicast messages the application AUTH_DATA is either the first application AUTH_DATA in the message or chosen by the PDP.

6. Message Processing Rules

6.1 Message Generation (RSVP Host)

An RSVP message is created as specified in [RFC2205] with following modifications.

1. RSVP message MAY contain multiple AUTH_DATA policy elements.

2. Authentication policy element (AUTH_DATA) is created and the IdentityType field is set to indicate the identity type in the policy element.

   - DN is inserted as POLICY_LOCATOR attribute.

   - Credentials such as Kerberos ticket or digital certificate are inserted as the CREDENTIAL attribute.

3. POLICY_DATA object (containing the AUTH_DATA policy element) is inserted in the RSVP message in appropriate place. If INTEGRITY object is not computed for the RSVP message then an INTEGRITY object SHOULD be computed for this POLICY_DATA object, as described in the [POL_EXT], and SHOULD be inserted as a Policy Data option.

6.2 Message Reception (Router)

RSVP message is processed as specified in [RFC2205] with following modifications.

1. If router is not policy aware then it SHOULD send the RSVP message to the PDP and wait for response. If the router is policy unaware then it ignores the policy data objects and continues processing the RSVP message.

2. Reject the message if the response from the PDP is negative.

3. Continue processing the RSVP message.

6.3 Authentication (Router/PDP)

    1. Retrieve the AUTH_DATA policy element. Check the PE type field and
       return an error if the identity type is not supported.

    2. Verify user credential

       - Simple authentication: e.g. Get user ID and validate it, or get
         executable name and validate it.

       - Kerberos: Send the Kerberos ticket to the KDC to obtain the
         session key. Using the session key authenticate the user.

       - Public Key: Validate the certificate that it was issued by a
         trusted Certificate Authority (CA) and authenticate the user or
         application by verifying the digital signature.

7. Error Signaling

   If PDP fails to verify the AUTH_DATA policy element then it MUST
   return policy control failure (Error Code = 02) to the PEP. The error
   values are described in [RFC 2205] and [POL-EXT]. Also PDP SHOULD
   supply a policy data object containing an AUTH_DATA Policy Element
   with A-Type=POLICY_ERROR_CODE containing more details on the Policy
   Control failure (see section 3.3.4). The PEP will include this Policy
   Data object in the outgoing RSVP Error message.

8. IANA Considerations

   Following the policies outlined in [IANA-CONSIDERATIONS],
   authentication attribute types (A-Type)in the range 0-127 are
   allocated through an IETF Consensus action, A-Type values between
   128-255 are reserved for Private Use and are not assigned by IANA.

   Following the policies outlined in [IANA-CONSIDERATIONS],
   POLICY_LOCATOR SubType values in the range 0-127 are allocated
   through an IETF Consensus action, POLICY_LOCATOR SubType values
   between 128-255 are reserved for Private Use and are not assigned by
   IANA.

   Following the policies outlined in [IANA-CONSIDERATIONS], CREDENTIAL
   SubType values in the range 0-127 are allocated through an IETF
   Consensus action, CREDENTIAL SubType values between 128-255 are
   reserved for Private Use and are not assigned by IANA.

9. Security Considerations

   The purpose of this memo is to describe a mechanism to authenticate
   RSVP requests based on user identity in a secure manner. RSVP
   INTEGRITY object is used to protect the policy object containing user
   identity information from security (replay) attacks. Combining the
   AUTH_DATA policy element and the INTEGRITY object results in a secure
   access control that enforces authentication based on both the
   identity of the user and the identity of the originating node.

   Simple authentication does not contain credential that can be
   securely authenticated and is inherently less secured.

   The Kerberos authentication mechanism is reasonably well secured.

   User authentication using a public key certificate is known to
   provide the strongest security.

10. Acknowledgments

   We would like to thank Andrew Smith, Bob Lindell and many others for
   their valuable comments on this memo.

11. References

   [ASCII]   Coded Character Set -- 7-Bit American Standard
                           Code for Information Interchange, ANSI X3.4-
                           1986.

   [IANA-CONSIDERATIONS]   Alvestrand, H. and T. Narten, "Guidelines for
                           Writing an IANA Considerations Section in
                           RFCs", BCP 26, RFC 2434, October 1998.

   [POL-EXT]   Herzog, S., "RSVP Extensions for Policy
                           Control", RFC 2750, January 2000.

   [POL-FRAME]   Yavatkar, R., Pendarakis, D. and R. Guerin, "A
                           Framework for Policy-based Admission Control
                           RSVP", RFC 2753, January 2000.

   [RFC 1510]   Kohl, J. and C. Neuman, "The Kerberos Network
                           Authentication Service (V5)", RFC 1510,
                           September 1993.

   [RFC 1704]   Haller, N. and R. Atkinson, "On Internet
                           Authentication", RFC 1704, October 1994.

   [RFC 1779]   Killie, S., "A String Representation of
                         Distinguished Names", RFC 1779, March 1995.

   [RFC 2205]   Braden, R., Zhang, L., Berson, S., Herzog, S.
                         and S. Jamin, "Resource ReSerVation Protocol
                         (RSVP) - Version 1 Functional Specification",
                         RFC 2205, September 1997.

   [RFC 2209]   Braden, R. and L. Zhang, "Resource ReSerVation
                         Protocol (RSVP) - Version 1 Message Processing
                         Rules", RFC 2209, September 1997.

   [UNICODE]   The Unicode Consortium, "The Unicode Standard,
                         Version 2.0", Addison-Wesley, Reading, MA,
                         1996.

   [X.509]   Housley, R., Ford, W., Polk, W. and D. Solo,
                         "Internet X.509 Public Key Infrastructure
                         Certificate and CRL Profile", RFC 2459, January
                         1999.

   [X.509-ITU]   ITU-T (formerly CCITT) Information technology -
                         Open Systems Interconnection - The Directory:
                         Authentication Framework Recommendation X.509
                         ISO/IEC 9594-8

12. Author Information

    Satyendra Yadav
    Intel, JF3-206
    2111 NE 25th Avenue
    Hillsboro, OR 97124

    EMail: Satyendra.Yadav@intel.com

    Raj Yavatkar
    Intel, JF3-206
    2111 NE 25th Avenue
    Hillsboro, OR 97124

    EMail: Raj.Yavatkar@intel.com

    Ramesh Pabbati
    Microsoft
    1 Microsoft Way
    Redmond, WA 98054

    EMail: rameshpa@microsoft.com

    Peter Ford
    Microsoft
    1 Microsoft Way
    Redmond, WA 98054

    EMail: peterf@microsoft.com

    Tim Moore
    Microsoft
    1 Microsoft Way
    Redmond, WA 98054

    EMail: timmoore@microsoft.com

    Shai Herzog
    IPHighway, Inc.
    55 New York Avenue
    Framingham, MA 01701

    EMail: herzog@iphighway.com

13. Full Copyright Statement

Acknowledgement