The John the Ripper software supports many operating modes, including a dictionary attack using an input wordfile. Several relevant passwords (Simson, SIMPSON, simson, HARVARD, Harvard, harvard,CSE-170, security, SECURITY, etc.) were added to the beginning of the wordfile. Running the software with wordfile resulted in the password being identified in a matter of seconds.

Imagine that you are a security officer working for a large government agency. You need to need to move large video files over the Internet to agents in the field. Assuming that the agents can successfully authenticate themselves to your system, design a protocol that allows them to transfer the video securely over the Internet. What sort of encryption do you want to use? There are two caveats: This is highly sensitive information, so many other governments may try to crack your system. Due to software validation constraints, once the protocol is approved, it cannot be changed for the next 5 years. Write a page describing a protocol to solve this problem and why you decided to use various types of security/encryption mechanisms.

It is assumed that the general system infrastructure will be comprised of the following major components: *video files* (pre-recorded video content files that are viewed by the field agents), *video servers* (computer systems that store and send video files to video clients), *video clients* (systems and applications used by field agents to view video files), and the *Internet* (unsecured communications network over which the video files are transmitted). It is also assumed that the established authentication processes are sufficient to confirm the identify of the users of the video clients (to the video servers), and the identify of the video servers (to the users of the video clients), including certificate verification.

To provide confidentiality for the video file data during transit, a hybrid end-to-end encryption scheme using public/private keys and a one-time symmetrical session key is proposed. The video file content will be encrypted with a random one-time session key using the AES algorithm (Advanced Encryption Standard, a US Government standard). The one-time AES session key will be encrypted with the RSA algorithm (named after its developers: Rivest, Shamir, Adleman).

The AES algorithm was selected because, according to the US Government, the AES algorithm with 128-bit and greater key lengths is adequate to protect (SECRET) classified information. A 128-bit key will be sufficient to encrypt the video file content, especially since data for each separate video viewing session will be encrypted with a random and temporary one-time session key (used only for a given session between a video client and a video server).

The RSA asymmetric public-key cryptography algorithm will be used for the encryption of the one-time AES session keys. RSA was selected because it is known to be reliable against cryptanalysis attack methods, and is a standard for public-key cryptography. A 2048-bit key size will be used for enhanced security during the distribution of the one-time session key. While this large key size is computationally intensive, is it only being used for the encryption of a small amount of data (the 128-bit AES key), so it should not excessively degrade system performance. It should also be safe from being cracked for the upcoming years.

This hybrid end-to-end encryption scheme works as follows:

1. After authentication, a video client sends a video request message to a video server. This message is encrypted with the video server's public key.

2. Assuming the video client is authorized to view the requested video file, the video server generates a random session key and encrypts it using the video client's public key.

3. The video server sends a message with the encrypted session key to the client. The video client uses its private key to decrypt message containing the random session key.

4. The video server reads data from the requested video file and encrypts it using the random session key. The video server then sends the encrypted video file data to the video client.

5. The video client receives the encrypted video file data, and decrypts it using the random session key, which was already made available. From there the video client displays the video content to the field agent.