inverses modulo p-1 using the Euclidean algorithm, so that decoding the subliminal message is easy in this case. If w is equally likely to take on any value, $0 \le w < p$, then the probability that S_X will be able to recover k in this way is $\varphi(p-1)/(p-1)$. We therefore have a situation in which $\varphi(p-1)$ messages—the $\varphi(p-1)$ positive residues modulo p-1 that are relatively prime with respect to p-1—can be encoded in the subliminal channel, with a probability of at most $\varphi(p-1)/(p-1)$ that the message can be recovered by S_X using the simple decoding rule. This is the subliminal channel that Simmons described in [7].

There are several points about this channel that need to be discussed—some having to do with security and some with concealment (of the fact that subliminal communications are occurring). Let M_p be the set of $\varphi(p-1)$ positive residues modulo p-1 that are relatively prime with respect to p-1. Since $k \in M_p$, the subliminal channel capacity is at most $d = \log_2|M_p| = \log_2(\varphi(p-1))$ bits. There is also the nontrivial problem of devising an efficient coding/decoding algorithm mapping between d bit binary messages and the members of M_p. In order to concentrate on only those aspects important to subliminal communication we will assume that such an algorithm is available and refer to messages (d bit binary sequences) and the corresponding session keys k, $k \in M_p$, used to communicate them as though they were the same thing.

If a public receiver knows the message, k', that the T_X wishes to communicate to the S_X and suspects that a subliminal channel of the sort described above is in use, he can easily verify his suspicions by calculating $\alpha^{k'}$. If $\alpha^{k'} = r$, he can be virtually certain that he has detected a subliminal communication. One way the T_X/S_X can conceal this is analogous to the way Vernam encryption was used to conceal the subliminal channel in the one-bit example described in the introduction. In this case, the T_X and S_X share in secret in advance of their use of the subliminal channel a one-time key[1] consisting of a sequence of randomly generated binary bits. When the T_X wishes to communicate a d bit message to the S_X, he first Vernam encrypts it with the next d bits of unused key, encodes the resulting d bit cipher into a member of M_p which is the "session key" k" that he uses in the El Gamal protocol. In this case, a public receiver, even though he knows the message that the T_X wishes to communicate to

1. For the statements about unconditional concealment and unconditional security made throughout this paper about various subliminal channels to hold, a true one-time key must be used. For practical purposes the T_X and S_X would simply share the key to any cryptographically secure key stream generator—the synchronized output of which would be used instead of the one-time key sequence described here.