Report on the Second Modes of Operation Workshop

The National Institute of Standards and Technology (NIST) sponsored a public workshop for the analysis of block cipher modes of operation on August 24, 2001, at the Holiday Inn-Santa Barbara, in Goleta, California. This report summarizes the presentations and discussions at that workshop.

The slide presentations are available on the world wide web at the NIST modes home page, http://www.nist.gov/modes, as well as the submission papers for modes proposals, public comments, and other information on the modes of operation development effort.

1. Welcome and Overview

Morris Dworkin, a mathematician from NIST's Computer Security Division, served as the moderator of the workshop. He welcomed the attendees and introduced the other participants from NIST: Bill Burr, Elaine Barker, Aaron Nelson, and, at the registration table, Vickie Harris and Teresa Vicente. He reviewed the three current Federal Information Processing Standards (FIPS) that establish block cipher modes of operation. FIPS 81 specifies four confidentiality modes for use with the Data Encryption Standard (DES) as the underlying block cipher: the Electronic Codebook (ECB), Cipher Block Chaining (CBC), Cipher Feedback (CFB), and Output Feedback (OFB) modes. FIPS 113 specified an authentication mode of the DES that is essentially the Cipher Block Chaining-Message Authentication Code (CBC-MAC) mode; this mode was also discussed in an appendix of FIPS 81. FIPS 46-3 approved the seven modes of operation of Triple DES that are specified in ANSI X9.52.

With the advent of new block ciphers such as the Advanced Encryption Standard (AES), there was a need to update existing modes of operation, and an opportunity to develop new modes of operation. The draft NIST "Recommendation for Block Cipher Modes of Operation" updated the ECB, CBC, CFB, OFB, and CBC-MAC modes for use with any approved underlying block cipher algorithm, and specified an additional confidentiality mode, the Counter (CTR) mode. The recommendation had been out for public comment since the end of July; the comment period closed at the end of August. To discuss new proposals for modes of operation and related issues, NIST held the first public workshop for the analysis of block cipher modes of operation on October 20, 2000, in Baltimore, Maryland. In the interim, other new modes had been submitted, and some of the previous proposals had been revised.

Dworkin reviewed the agenda for the workshop. First, as in the first public workshop, several submitters of modes of operation would present their proposals, including three sets of proposals for authenticated encryption modes and four sets of proposals for authentication modes. NIST would briefly summarize each of the remaining modes that had been submitted. Second, a session would be devoted to technical comments on modes and their uses, including the presentation of one paper. Third, a session would be devoted to the discussion of comments that NIST had so far received on its draft

Recommendation. Fourth, a session would be devoted to the discussion of general issues and next steps.

2. Presentations of Modes of Operation

The presentations of three authentication modes (XCBC, RMAC, PMAC) were followed by the presentations of three sets of authenticated encryption modes (OCB; XCBC-XOR, XECB-XOR; IAPM, IACBC), the second presentation also included an authentication mode (XECB-MAC). NIST presented brief summaries of the five remaining proposals (2DEM, ABC, KFB, PCFB, AES-hash).

2.1 XCBC: A Version of the CBC MAC for Handling Arbitrary Length Messages

John Black, of the University of Nevada, spoke on the XCBC algorithm, which he and Phillip Rogaway had proposed in a paper at the Crypto 2000 conference. He first reviewed the general concept of message authentication via a message authentication code (MAC), and he explained the specifications of the CBC-MAC mode. The CBC-MAC algorithm was simple and widely used: it was standardized in ANSI X9.19, FIPS 113, and ISO 9797. However, the CBC-MAC algorithm was only specified for messages that were composed of complete blocks, and, for any given key, it should only be used to authenticate messages that consist of a constant number of blocks. He listed several previous variations of the CBC-MAC mode that handled messages with a variable number of blocks "correctly," i.e., achieving some standard security properties; the XCBC mode was their contribution to this list.

He presented the specifications of the XCBC mode. Its advantages included the following: it used a minimal number of block cipher invocations; it correctly handled messages of any bit-length; the block cipher was invoked with only one key, and only in the forward direction; the algorithm could be executed on-line (i.e., for example, it was not necessary to count the number of bits in the message in advance of computing the MAC). Moreover, the XCBC mode was easy and familiar to implement and patent-free. The two disadvantages were that the XCBC mode was not parallelizable—a property inherent in the CBC-MAC—and that the XCBC mode required more key bits than the CBC-MAC mode. He explained a standard construction for deriving the necessary key bits from a single block cipher key.

Another important advantage was that the XCBC mode was a pseudo-random function (PRF), which had cryptographic applications in addition to its use as a MAC algorithm. An automatic consequence was that the XCBC mode was "provably secure" under the assumption that the underlying block cipher was a pseudo-random permutation (PRP), i.e. under the "random oracle model". He elaborated on these notions: the security properties were essentially an upper bound on the adversary's chance of constructing a forgery in terms of the number of allowed "queries" to the MAC "oracle." In practical terms, an adversary had less than a one in a trillion chance of constructing a forgery even

after observing, for a month, a thousand messages (no larger than 10 Kbytes each) per second.

In response to a question from an attendee, Black observed that, because the XCBC mode, like the standard version of the CBC-MAC mode, did not require a nonce, nor any initialization vector (IV), it was up to the user to protect against replay attacks, for example, by employing time stamps.

2.2 RMAC: A randomized CBC-MAC beyond the Birthday Paradox Limit

Antoine Joux, of the DCSSI Crypto Lab, presented the RMAC mode, a randomized MAC mode. He began by reviewing the definition and properties of the CBC-MAC mode, including the "classical" padding of messages to a bit length that was a multiple of the block size. As in Black's talk, Joux discussed why the CBC-MAC mode was not appropriate for encrypting messages with variable numbers of blocks; he gave specific example of the construction of a "forgery," i.e., message and its correct MAC tag, without knowing the secret key of the block cipher.

Before presenting RMAC, he presented DMAC, a variation of CBC-MAC in which the new MAC tag is produced by encrypting the old MAC tag under a second key. This variation was provably secure in the random oracle model. However, the given security bound was unsatisfactory in the following sense: it would very likely be easier, i.e., require a far smaller number of actions, to construct a forgery by querying an oracle than it would be to find the secret key by exhaustive search. In particular, a mathematical principle, known as the "birthday paradox," dictated that if the MAC tag output had t bits, only $2^{t/2}$ queries to the MAC oracle would likely result in the discovery of two messages with the same tag, i.e., a "collision," from which forgeries could easily be constructed. Thus, if the key had the same number of bits as the tag output, then the number of queries needed to produce a forgery would be expected to be only the square root of the number of keys.

He mentioned the deficiencies of three existing solutions to this concern: MACRX was not based on CBC MAC and tripled the size of the tag; the use of counters required the maintenance of state, and a certain straightforward randomized method was still vulnerable to a type of collision.

The RMAC algorithm was a refinement of the DMAC algorithm in which a random bit string was exclusive-ORed into the second key and then appended to the resulting MAC to form the tag. The birthday paradox in principle was no longer relevant, for, say, the AES with 128 bit keys, because the tag would be doubled to 256 bits. Joux presented his underlying security model and the properties that he had proven for RMAC: the number of queries that bounded the chance of a forgery was relatively close to the number of 128 bit keys.

The performance of the RMAC mode was optimal in two senses. First, it required the same number of key bits and block cipher invocations as DMAC. Second, the doubling of the MAC size compared to CBC-MAC or DMAC was the minimum increase that would allow the birthday paradox to be circumvented.

An attendee raised the concern that, in practice, it is often difficult/expensive to implement a good source of random bits.

2.3 PMAC

Phillip Rogaway, of the University of California at Davis, presented the PMAC mode, which stands for parallelizable message authentication code. He observed that the CBC-MAC mode was inherently serial; one goal of the PMAC mode was to allow parallelizability without much cost in efficiency compared to the serial calculation. Other improvements over the CBC-MAC mode were to allow messages of any bit length and to operate "correctly" across messages of varying lengths.

Before presenting the specification, he discussed PMAC's properties. PMAC was a variable input length PRF, stateless and deterministic, and fully parallelizable. It required essentially one call to the underlying block cipher per input block, plus one block cipher call as a session-startup, and only one block cipher key. Its other required operations were exclusive-OR and shifts, but not modular additions. After presenting the specification, he listed some related work.

He presented performance data that showed PMAC to be only 8% less efficient than CBC-MAC, and that suggested that the quality of the code might be as important as algorithmic differences in measured timings.

He explained the notion "provable security" as the following kind of "reduction": if an adversary could attack the mode, then an adversary could also construct an attack on the underlying block cipher. Provable security theorems quantified the loss of security across the reduction. He presented the specific theorem that held true for PMAC.

He concluded by comparing the PMAC mode to other MAC algorithms (CBC-MAC, XCBC, and three versions of XECB-XOR) across the following properties: input domain, status as a PRF, the length of the MAC tag, parallelizability, the number of block cipher calls, and the number of overhead computations per block.

2.4 OCB Mode

Rogaway also presented the OCB mode, an authenticated encryption mode. Combining the services of authentication and privacy into a single cryptographic primitive was more efficient and easier to use correctly than addressing the two services separately.

He presented a hierarchy of security notions, including "indistinguishability under chosen plaintext attack" versus "indistinguishability under chosen ciphertext attack." The OCB

mode achieved the latter notion, which was stronger, i.e., more secure, than the former. In fact, in the public key setting, the cryptographic community seemed to demand the stronger notion, although in the symmetric key setting, modes that achieved only the weaker notion, such as the CBC and CTR modes, were generally accepted. (He mentioned that all of the notions were predicated on an idealized model of the block cipher as random permutation.) He presented an example of the use of an encryption primitive within a handshake protocol that was incorrect under the weaker property but correct under the stronger notion. Because protocol designers did not necessarily understand the issue, it was prudent for primitives to satisfy the stronger notion of security.

He discussed how (separate) privacy and authentication primitives were typically combined today, and the advantages and disadvantages of this approach. He listed previous attempts to combine the services into one primitive; Jutla was the first to propose a "correct" scheme, in August, 2000. The OCB mode was inspired by Jutla's IAPM but had many new characteristics.

He summarized the characteristics of the OCB mode. It was an authenticated encryption scheme that used any block cipher and a single block cipher key. It applied to inputs of any length and returned a ciphertext of minimal length. It was good in both hardware and software, and its computational cost was comparable to the CBC mode, except that OCB was fully parallelizable, with two additional calls to the block cipher. Its nonce did not need to be unpredictable; it was endian-neutral; it did not require *n* bit addition; and it had quick key setup, suitable for single-message sessions. It was provably secure, and it was being considered in the IEEE 802.11 draft. He presented the specifications of the mode.

He then presented performance data for OCB, for both assembly code and C code, on a Pentium 3 processor, in both cases with only a slight performance degradation compared to CBC encryption.

In his discussion of the provable security of the OCB mode, he asserted the importance of properly defining and proving security properties. As an illustration, he cited his knowledge of a good definition for the security goals as an important aspect of his ability to quickly construct an attack on a different authenticated encryption scheme, the Dual Counter Mode. He then presented the reduction theorems for the OCB mode, with respect to both privacy and authenticity.

He discussed the usefulness of provable security: it provided strong evidence that a mode did what it was intended to do; it was the best assurance that the cryptographers knew how to deliver; and, it gave quantitative guidance in the usage of the mode. Provable security, did not, however, provide an absolute guarantee of security. In particular, some security issues were not necessarily captured by the abstractions of the models, and it did not provide protection against usage errors and implementation errors.

He presented a chart that compared the three sets of authenticated encryption modes that would be presented at the workshop: the OCB mode, the IAPM mode, and the XECB-

XOR modes. The following elements of the modes were compared: the input domain, the bit length of the ciphertext, the IV requirements, the number of calls to the block cipher for each message and for each key setup, the key length and the number of encryption keys, the per block computational overhead (i.e., other than the block cipher call), and the encryption circuit depth.

In response to an attendee's question, he elaborated on the encryption circuit depth. The OCB mode would require that three of the block cipher calls be performed in series. He had not worked to minimize this element because he did not think it was very important in practice. He observed that reducing this depth would result in other kinds of costs.

Another attendee inquired about three potential additional security services. Rogaway replied that the OCB mode could be used purely for encryption simply by omitting the tag; however, it could not be used purely for authentication by omitting the ciphertext, and it could not be used as a commitment scheme (by keeping the key secret from the receiver) unless the security assumptions on the block cipher were upgraded. A fourth application was possible: a pseudo-random bit generator.

A third attendee pointed out that it was conventional in network protocols for some data in each message to require only authentication while other data required authentication and privacy. Rogaway responded that he had considered this, and that he would soon come out with a paper that describes a proper extension of the OCB mode that addressed this need, in a manner that is almost free.

A fourth attendee asked if it was possible to provide reduced strength encryption with full strength authentication (presumably with better efficiency than full strength encryption); Rogaway did not know how to provide that functionality with the OCB mode.

2.5 The XCBC-XOR, XECB-XOR and XECB-MAC Modes

Virgil Gligor, of VDG Inc., presented two authenticated encryption modes, called the XCBC-XOR and XECB-XOR modes, and one authentication mode, the XECB-MAC modes. He first mentioned some of his years of experience working on the difficult problem of constructing authenticated encryption modes.

He discussed a framework for analyzing modes. He defined a security claim as a security notion that was supported by a mode or scheme of encryption, where a security notion was a goal coupled with a style of attack. For example, a good notion of privacy was indistinguishability under adaptive chosen plaintext attacks, and a good notion of authenticity was protection against existential forgery under (adaptive) chosen plaintext attacks. The wish was for his modes to achieve these notions in both single key systems and also in two-key systems, where confidentiality is separated from integrity.

He then defined an operational notion as an operational goal coupled with certain modes characteristics. He discussed three categories of operational goals. 1) Cost-performance included power consumption, the number of block cipher calls, latency (what Rogaway

called encryption circuit depth), and implementation cost, such as chip area in hardware. 2) Simplicity was reflected in several aspects: the use of a single key, in the specification, (particularly the operations used), and the use of the same basic structure for different purposes. 3) Usability in different environments included key-state protection mechanisms, the availability of random number generators, and error recovery requirements.

He discussed the following categories of modes characteristics: state, degree of parallelism, error recovery, separation of confidentiality and authentication, and padding. He pointed out that his padding scheme was standard except that it did not require an entire block of padding when messages fell on the block boundary; the scheme did not degrade the security bound, while avoiding the need for a method like ciphertext stealing, which inherently required extra latency and an extra block cipher encryption.

His modes provided a three options with regard to the maintenance of state: stateful, stateful sender, and stateless. He discussed these options in terms of the tradeoffs between the performance provided by the use of state and the robustness provided by avoiding state.

He described his original framework for the specification of his authenticated encryption modes. The framework called for 1) an encryption mode 2) a set of unpredictable elements to combine with the outputs of the mode, and 3) an invertible operation for combining these elements, such as exclusive-OR or modular addition. The mode was assumed to provide indistiguishability under chosen plaintext attack, and the underlying block cipher had to process the input data itself, as opposed to, say a counter, as occurred in the CTR mode. The result of applying the inverse of the operation to any pair of the unpredictable elements also had to be unpredictable. (He mentioned that Dual Counter Mode did not satisfy this condition, which led directly to an attack.) A special case of the original framework using the CBC mode had been invented in late 1979. He explained that the framework could be used for plain authentication as well as authenticated encryption and encryption; although such authentication standing alone might not be the most efficient method, the marginal cost of adding this functionality in hardware was just some control circuitry.

He explained the three methods of mode initialization. The mode initialization in the stateless case could be achieved using random number generators; in the stateful sender case, it could be achieved by encrypting counters as a source of random numbers; in the stateful case, two random variables could be used, along with a counter, and the resulting sequence of unpredictable values could be reused for different messages.

He then presented specifications of his modes proposals. He presented the family of three XCBC-XOR modes together, each of the three XECB-XOR modes separately, the segmented stateful sender mode, and the stateful XECB-MAC. The XECB-XOR modes were similar to Jutla's IAPM mode in structure; however, the unpredictable elements were not pairwise independent, at the cost of a small degradation in security bounds (by a fraction of a logarithmic factor), the maximum efficiency in block cipher invocations and

latency could be achieved. Unlike the OCB mode, the generation of the unpredictable elements was designed for parallel execution.

He summarized his presentation, and then mentioned some points regarding the history of his modes. He downplayed the role of the XOR-MAC as only a starting point for the design of the XECB-MAC mode. The XCBC-XOR was proposed in late 1999 and submitted to the patent office in January, 2000, and distributed to some members of the cryptographic community in February, 2000. The XECB-MAC plus some new variants of the XCBC-XOR were sent to the patent office in March, 2000. The XECB-XOR modes were sent to the patent office in August, 2000.

An attendee asked whether his patents applied to other algorithms presented at the workshop. Gligor thought that his patents might apply to the IAPM modes and the OCB modes, but this was really a question for others to address, e.g., courts and judges.

2.6 IAPM

Charanjit Jutla, of the IBM T.J. Watson Research Center, presented the Integrity Aware Parallelizable Mode (IAPM). He first introduced the CBC mode and the CBC-MAC mode. He explained how the natural attempt to combine these modes was vulnerable to forgeries by rearranging the blocks of the ciphertext blocks. He presented a variant with some additional forward chaining that was proven secure, but the scheme required too many initialization blocks.

The first step toward IAPM was to realize that, instead of the additional forward chaining, the CBC output blocks could be "whitened," i.e., exclusive-ORed, with a sequence of random values. It was expensive (in terms of block cipher invocations) to generate a fully independent sequence of random values; however, for the same level of security it sufficed to use a pairwise independent sequence, which required far fewer block cipher invocations. In fact, a "pairwise differentially uniform" sequence was sufficient, and even cheaper to construct. The result was the Integrity Aware CBC (IACBC) mode, which used a subset construction to convert the results of roughly log m invocations of the block cipher on incremented IV values to form a sequence of m pairwise independent output whitening values.

He then presented the IAPM mode, a refinement of IACBC in which the CBC chaining was replaced by input whitening taken from the same sequence of random values as was used for the output whitening. The subset construction was not the only possibility for generating the sequence of whitening values: he presented an algebraic construction that required only two block cipher invocations to yield a pairwise independent sequence, and a second algebraic construction that required only one block cipher invocation to yield a pairwise differentially uniform sequence.

He discussed the provable security bounds of the IAPM mode. With respect to privacy, the bounds were of the same order as the bounds for the CBC mode, and with respect to message integrity the bounds were the same as the bounds for the CBC-MAC mode.

Moreover, he had shown that if the pairwise differential uniform sequence was only approximated, then the privacy bounds would degrade accordingly. He sketched another security result, namely, if an authenticated encryption scheme was constructed only using the exclusive-OR operation and block cipher invocations, then it was impossible to achieve equivalent security bounds as IAPM with fewer extra block cipher invocations than were required by his subset construction.

He presented examples of constructions of random sequences, and the advantages and disadvantages of each construction. The first method he discussed was the "$i*a$ mod p" construction. It required only a single extra encryption and a single extra word of storage in both serial and parallel implementations, with no worry about buffer overflows or pointer implementations. In serial implementations, the updating of the successive random values was fast. For parallel implementations, a full hardware multiplier was not required to multiply a 128 bit value by a small integer. He explained a direct method for performing such calculations that was easy to handout to parallel processors. The requirement for 128 bit addition was a minor disadvantage.

The second method he discussed was subset construction, which had the advantage of only using the exclusive-OR operation. He explained how a Gray code could be used to efficiently regulate the updating of the successive random values. This updating was easy in serial implementations, and, in parallel, he claimed the calculation of the subset construction was also easy, although the handout to each processor was harder. He also identified two other disadvantage of the subset construction: the number of block cipher invocations that it required was logarithmic in the number of blocks in the message, and either a pointer implementation or a large, fixed allocation of RAM was required.

The third method he discussed was the "Galois field" construction. Like the $i*a$ mod p construction, it had the advantage of requiring only one block cipher invocation. Like the subset construction, a Gray code gave easy updating of the successive random values in serial implementations; in parallel, the handout to each processor was harder. He identified two other disadvantages: either a pointer implementation or a large, fixed allocation of RAM was required, and the left shifting of 128 bit words was required, which he regarded as comparable to 128 bit addition.

He presented software performance data for IAPM. Using the reference code for Rijndael on the AES website, on an IBM 200 MHz Power PC, the subset construction for IAPM yielded a throughput of about 43.5 Mbits per second, when applied to messages of 1024 blocks. Plain CBC mode, which did not authenticate the message like IAPM, performed only slightly better, about 46.8 Mbits per second.

He discussed the hardware performance of IAPM, in particular a new hardware design of Rijndael that IBM had presented at the CHES 2001 conference on cryptographic hardware. The parallelism of IAPM allowed for a relatively slow clock speed, which in turn allowed greater gate depth and a much smaller gate count. Another savings was in the hardware area that was devoted to subkey storage: in a parallel implementation, essentially one subkey was required at a time; by contrast, pipelined implementations

required the simultaneous storage of every subkey. These area savings allowed for a greater number of encryption engines. Thus, while previous hardware implementations of Rijndael had achieved throughput on the order of 1 Gbit per second, the IBM IAPM implementation had achieved 7.5 Gbit per second, almost a terabit per second.

Comprehensive test vectors had been provided to NIST and posted on the NIST website.

IBM had filed patents on "all these schemes." If one of the schemes was standardized, IBM would license them on a non-discriminatory, non-exclusive basis and at a reasonable rate.

An attendee raised the following two issues with respect to the IPSEC protocol. First, there were messages in which part of the data requires authentication but not encryption; Jutla did not immediately see how the authentication service could be separated from the encryption service in this context, because the integrity was, in a sense, spread throughout the ciphertext. The second observation was that the merits of the parallel approach versus pipelining might not be so clear when there were multiple data streams that were encrypted under different keys. Jutla mentioned that he had proposed an Internet Draft for the use of IAPM in IPSEC.

2.7 Remaining Modes

As a service to the modes submitters that could not attend the workshop, Aaron Nelson, a summer employee at NIST, presented the specifications of the remaining five modes, along with brief summaries of their properties.

The Two Dimensional Encryption Mode (2DEM) was a confidentiality mode submitted by Ahmed A. Belal and Moez A. Abdel-Gawad. The mode required an secret parameter, called *BPR* (blocks per row) in addition to the key. The plaintext was arranged into an array of bytes, in a manner regulated by the parameter, and the block cipher was invoked in two passes: first the rows were encrypted, and then the columns of the resulting array were encrypted. The authors claimed that the mode was well suite for the encryption of images, and that the interleaving provided resistance to certain attacks.

The Accumulated Block Chaining Mode (ABC) was a confidentiality mode submitted by Lars Knudsen. In addition to the chaining of the CBC mode, the ABC mode incorporated the chaining of a separate set of values into both the input and output of the block cipher. These separate values were obtained iteratively by the application of a suitable accumulation function to successive plaintext values. The ABC mode required two IVs. The ABC mode had the property of infinite error propagation, but the author did not claim that authentication was achieved. The infinite error propagation provided extra diffusion, so that the mode functioned somewhat more like an giant block cipher than the CBC mode. The ABC also better resisted birthday attacks.

The Key Feedback Mode (KFB) was a random bit generator submitted by Johann Håstad and Mats Näslund. The block cipher was first invoked on the first plaintext input block

under the given key; each successive block cipher output block was then used as the key for the next invocation of the block cipher. A matrix of random bits was applied to each output block to generate the desired stream of bits. The mode required an initial matrix and a constant in addition to the initial key. The purpose of the construction was to obtain a stream of bits that was provably pseudo-random without assuming that the underlying block cipher acted as a PRF. Instead, the mode relied on the weaker assumption that one or more iterations of the block cipher used in the prescribed manner, i.e., with varying keys and fixed plaintext, would be difficult to invert.

The Propagating Cipher Feedback Mode (PCFB) mode was a confidentiality mode, with a variant for authentication, that was submitted by Henrick Hellström. The mode was similar to the CFB mode, except that the input blocks (after the IV) consisted of the previous ciphertext fed back into the previous output block, rather than the previous input block. As a result, the PCFB mode offered two-way error propagation.

The AES-hash mode was a hash mode that was submitted by Bram Cohen and Ben Laurie. It was a variant of the Davies-Meyer hash construction with an additional block cipher invocation at the end, in order to prevent an adversary from easily obtaining the hash of an extension of a message whose hash was known. An attendee pointed out that the specification called for the variant of Rijndael, requiring a block size of 256 bits, that was not yet planned for standardization. Nelson acknowledged that, for this reason, the AES-hash mode would not be adopted in the current development effort.

3. Technical Comments on Modes of Operation

Matt Blaze, of AT&T Labs Research, presented a short talk on the paper "Cryptographic Modes for the Internet" that he had co-authored and submitted as a public comment. He wanted to communicate two main points.

First, even the most aggressive attack models that are considered in the development of modes should be taken seriously. The Internet involved the delivery of datagrams across a potentially unreliable network. The types of attacks that the cryptologic community might consider only in aggressive attack models, such as adaptive chosen plaintext attacks or adaptive chosen ciphertext attacks, could be readily set up in the normal operation of the Internet. He discussed several vulnerabilities: the network might duplicate packets; packets might be delivered out of order or lost altogether; packets can be misdirected to computers they were not intended for; and servers often acted as oracles for encrypt and potentially for decrypt operations. The ease of forging and capturing traffic was well known. Moreover, depending on the layer at which encryption was implemented, a single key could end up protecting several screens that had mutually untrusting users who have complete control of the data. The bottom line was that fancy attacks were not just theoretical.

Second, he urged NIST to include extensive guidance in the specification of the modes: what is required of the environment in which the mode is used, what is assumed about the applications, what security properties a mode can and cannot be assumed to have, and

specific examples, both positive and negative. He discussed several examples of security vulnerabilities resulting from incorrect implementations and protocols resulting from the misuse or misunderstanding of modes and their underlying assumptions. For example, the ability to maintain crypto-state could not be taken for granted, which had implications for the management of IVs and key. In particular, keystream modes like the OFB and the CTR modes were extraordinarily difficult to use correctly in most Internet applications below the reliable transport layer. One aspect, illustrated by a comment from an attendee, was that there was often pressure to use manual keying, under which the keystream modes were not as resilient as other modes.

After the presentation of the paper, there were no further technical comments on the modes.

4. Selected Comments on the Draft NIST Recommendation

Dworkin began a discussion of the draft NIST Recommendation for block cipher modes of operation by thanking the people who had already submitted public comments; moreover, public comments would continue to be accepted for another week, until August 31, 2001. Meanwhile, NIST wanted to consult with the workshop attendees in addressing some of the larger issues that were raised by the received comments.

Before the discussion of specific issues, Dworkin mentioned three general considerations. First, there was usually a tension between restricting options for security and interoperability, and providing more options for flexibility and for the accommodation of current practices. Second, the Recommendation was meant to apply to current and future approved block ciphers: were there issues for which the Triple DES modes ought to be handled differently than the AES? Third, the AES FIPS might be approved within the next several weeks, and NIST wanted to issue the Recommendation around the same time, so it was desirable to choose the simplest workable solutions.

Several commenters had raised an issue that had already been discussed in some of the workshop presentations: the CBC-MAC mode, when applied to messages consisting of different numbers of blocks under a single key, was vulnerable to existential forgeries of extensions of messages with known MACs. NIST recognized the impracticality of the requirement in the current draft to restrict the use of the mode for a given key to messages consisting of a constant number of blocks. Dworkin listed several other possible solutions to the problem: use the XCBC mode or the RMAC mode, double/triple encrypt the final CBC-MAC block, prepend the message with its length in blocks, truncate the MAC, or warn the user of the possible consequences of using the CBC-MAC mode for messages of different lengths. The consensus of the attendees was for NIST to provide a menu of options. This was the approach of other standards on this question, and in general was appropriate for modes to provide flexibility of use.

Also raised in earlier workshop presentations was the extent to which the modes should apply to messages whose bit length is not a multiple of the block size. The OFB and CTR modes already applied to messages of any length. There was some sentiment in

favor of including “ciphertext stealing” as an option for the CBC mode, and of providing a menu of suggested padding options. There was no comment on whether the proposed XCBC mode should be offered as an option of the CBC-MAC mode in order to address this concern.

The level of guidance in the document was commented upon in several contexts. Dworkin suggested that NIST might develop a separate guidance document later in order to expedite the publishing of the Recommendation. At least one attendee supported this idea, but another was concerned that, without the impetus of the specification document, the guidance document might not get written quickly or at all. In general, the attendees favored the inclusion of as much guidance as possible in the selection and use of the modes. Attendees supported the specific comment that NIST should deprecate the use of the ECB mode; it was also suggested that the OFB and CFB modes were somewhat outdated.

With respect to the appropriate properties of IVs, there was no objection to adding the requirement that the IVs for CBC and CFB modes to be unpredictable by an adversary; moreover, invocating the block cipher on the IV should be given as an option for achieving this property. There was no comment on the possible need for requirements on the IV for the OFB mode.

There was no support for the restriction of the possible segment lengths in the CFB mode to a small subset of values, such as {1, 8, 128}. An attendee suggested that NIST should provide examples for at least one odd segment length; however, NIST need not provide them for every segment length as suggested in one in the sets of comments.

There was no objection to the comment to promote the truncation of the CBC-MAC tag by four bytes as a general default. There was some support for NIST’s inclination to maintain 32 bits as the lower limit on the size of the tag, despite the suggestion that, in some specialized situations, an 8 bit tag might be appropriate.

An attendee agreed with NIST’s inclination not to include interleaved modes as suggested in one of the sets of comments.

5. Where do we go from here?

Bill Burr, Group Leader of the Security Technology Group of NIST’s Computer Security Division, gave a presentation on general issues and the next steps of the process. Problems with the current DES modes were obvious: they did not apply directly to the AES, only the ECB mode was fully parallelizable, and there has been cryptographic progress since the DES modes were specified. For example, there were now modes with strictly defined security properties, authenticated encryption modes, parallelizable modes, and modes for specific applications. NIST was taking the obvious steps to address the situation: updating the DES modes for use with any approved block cipher, adding the CTR mode, which was parallelizable, and holding workshops for the development of new modes.

The first issue he discussed was the users' needs. He contended that users wanted modes that were highly resistant to practical attacks. Proofs of properties were one way to ensure this, but failure to meet particular properties may not lead to practical attacks. As a caveat, he noted that attacks that are impractical today might be practical in a decade. Of course, performance, interoperability, and cost (including patent licenses) mattered to users too.

It was not clear even how many new modes to develop. One possible approach would be to accept every arguably useful mode that seemed to be secure; another approach might be to minimize the number of modes to promote interoperability, avoid potentially insecure or dangerous alternatives, while still providing reasonable coverage needs. Clearly, not all of the fourteen proposed modes were needed, but perhaps some other modes were needed, such as a super-encryption mode or a hash mode.

The form of the specification of modes had consequences. A FIPS was mandatory for users in the Federal Government, absent a waiver. This form was relatively inflexible: there was typically a five year change cycle. A Recommendation was more flexible to accommodate evolution, but it probably entailed more risks, since it was not mandatory. If a Recommendation was the initial form of a specification, then the timing of the transition to a more restrictive regime was an issue.

The level of implementation flexibility was another general issue. Restrictions limited utility; on the other hand, options limited interoperability. Options also increased the chances of mistakes, which could only partly be addressed by testing, as testing many options properly was more difficult. He observed that the following four levels of implementations: algorithm, mode, protocol, and application. The user only saw the application, and that was the only level at which the value or integrity of the data were known.

Protocol interactions presented another set of tradeoffs. Burr doubted that modes could be so comprehensive as to prevent such mistakes by protocol designers, but guidance could probably be improved. He suggested that a better solution might be the participation of more cryptographers in the development of protocol standards.

He raised the question of what categories of modes were needed, and how to weigh the relative importance of various properties. He listed the following categories of modes: authenticated encryption modes, authentication modes, hash modes, super encryption, or modes designed for specific applications. He listed the following properties to consider: performance (probably measured by the number of block cipher invocations), parallelizability, error expansion, synchronization, state, formal security properties, and intellectual property.

He observed that patented modes were very, very unpopular for many reasons. He recognized that people would generally urge NIST to prefer unencumbered modes to

encumbered modes, but absent a good alternative, should NIST refuse to standardize a patented mode if it offers a huge advantage?

He observed that, although modes last a long time, they are a rich, evolving subject, and he did not expect to be able to announce one or two modes and then leave them alone for two decades, as occurred with the DES modes. For example, the analysis of the modes and the protocols that used them was not simple. Therefore, he postulated that an ongoing process/approach was going to be necessary.

He offered the following ideas for next steps as a strawman. First, a modes document would be multi-part, with new parts added as modes became ripe, in consideration of new needs and issues. Second, NIST should consider the development of a separate guidance document on the use and selection of modes. Third, NIST should probably hold regular workshops or meetings to consider new modes. He opened the floor for discussion, comments, and advice.

An attendee explained some of the hostility of the cryptographic community to patented algorithms, and the IETF more generally, by its reliance on free and open-source software. Burr wondered whether, as a consequence, if NIST adopted a patented mode, the IETF would refuse to use it.

Another attendee suggested that NIST should buy the patents; Burr responded that this would be difficult.

An attendee commented that modes should be made available that remedy the problems that had been identified with the DES modes, and that were parallelizable to meet the demands of modern data rates.

An attendee observed that patented modes posed a disproportionate burden on small businesses and international users, and urged that no encumbered modes be adopted. (Later, another attendee claimed that this was not a disaster unless older modes were forbidden, which was not NIST's intention.) Burr acknowledged the difficulty, but he would not categorically rule out patented modes, especially since he was employee of the Department of Commerce, to which the patent office also belonged.

An attendee suggested that for high-speed (hardware) applications, a patented mode would be better than none, which would essentially leave implementers out on their own.

An attendee urged NIST to consider both hardware performance and software performance. Another attendee observed that the situation with respect to intellectual property costs of hardware versus software was subtle: he cited the case of Microsoft and Netscate buying licenses to use RSA and then giving away products that used it, while software developers were sometimes willing to pay licensing costs. The same attendee also argued that pipelined performance was often at least as important as parallel performance.

Several attendees spoke of the need for high speed, parallel authentication to go with the CTR mode.

An attendee observed that the authenticated encryption modes were not really appropriate for network protocols, because an approach based on universal hashing was an order of magnitude faster. Gligor pointed out that one example, the UMAC algorithm, was only asymptotically fast for large message sizes, but not as fast as the three proposals for authenticated encryption for messages smaller than 128K bytes, nor were such algorithms parallelizable in an architecture independent manner. The attendee pointed out that there were other universal hash functions besides UMAC. Burr wondered why UMAC had not been submitted to the development effort; Rogaway explained that it was a software-oriented algorithm and not necessarily suited to general environments, as it was highly parameterized and required a lot of time for key setup.

Burr asked for specific advice in selecting one of the three authenticated encryption modes. (He observed that, in the case of the AES, the cryptographic community had overwhelmingly rejected the idea of multiple algorithms.) He also asked whether the attendees considered the proposals sufficiently mature and the conditions ripe for standardization. One attendee reiterated the importance in some applications for the functionality of authenticating data that was associated with the encrypted data. Another attendee believed that standardization would be premature, because the intellectual property situation among the three sets of algorithms was still unsettled (patents were pending).