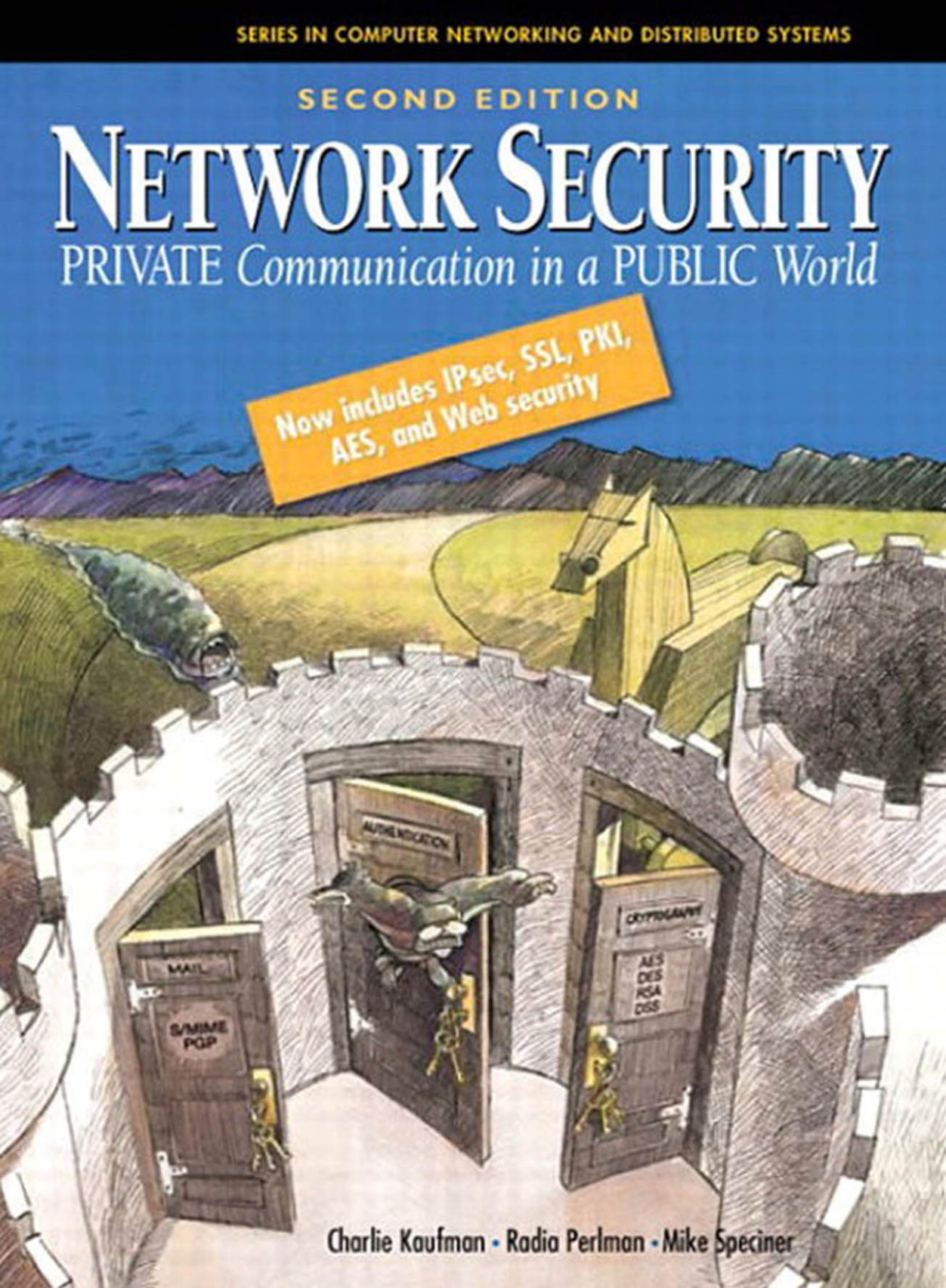
SERIES IN COMPUTER NETWORKING AND DISTRIBUTED SYSTEMS

SECOND EDITION

# NETWORK SECURITY

## PRIVATE *Communication in a* PUBLIC *World*

Now includes IPsec, SSL, PKI, AES, and Web security




Charlie Kaufman • Radia Perlman • Mike Speciner

# NETWORK SECURITY

PRIVATE Communication in a PUBLIC World

ISBN 0-13-046019-2

**Euler's Theorem Variant:** For all $a$ in $\mathbf{Z}_n$*, and any non-negative integer $k$, $a^{k\phi(n)+1} = a \bmod n$.

**Proof:** $a^{k\phi(n)+1} = a^{k\phi(n)}a = a^{\phi(n)k}a = 1^k a = a$.

The variant doesn't tell us that raising any number $m$ to the power $k\phi(n) + 1$ gets $m$ back mod $n$. It only works for $m$ in $\mathbf{Z}_n$*, which means $m$ must be relatively prime to $n$. It turns out that for numbers $n$ of the form we are interested in for RSA (the product of two primes), it is still the case that $m^{k\phi(n)+1} = m \bmod n$, even if $m$ was not relatively prime to $n$.

Do we really care? What is the probability that we'll ever find a number to encrypt that isn't relatively prime to $n$? (See Homework Problem 11.) Well, in case we care, RSA will still work even if the message is not relatively prime to $n$. And the proof isn't very hard.

### 7.8.1 A Generalization of Euler's Theorem

We will show that for numbers $n$ of the form used in RSA, namely $n = pq$, where $p$ and $q$ are distinct primes, $a^{k\phi(n)+1} = a \bmod n$ for all $a$ in $\mathbf{Z}_n$ (not just for $a$ in $\mathbf{Z}_n$*) as long as $k$ is a non-negative integer. We will use the Chinese Remainder Theorem.

If $a$ is relatively prime to $n$, the result is true by Euler's Theorem Variant. So our only problem is if $a$ is not relatively prime to $n$, which means $a$ is a multiple of $p$ or $q$. Let's say $a$ is a multiple of $q$. Let's compute the decomposed representation for $a^{k\phi(n)+1} \bmod n$. In other words, we want to find out what $a^{k\phi(n)+1}$ is mod $p$ and mod $q$. Then, by the Chinese Remainder Theorem, we'll know what it is mod $n$.

Since $a$ is a multiple of $q$, it must be relatively prime to $p$ (or else $a$ is a multiple of $n$, in which case it is 0 and the result is trivially true). Since $a$ is relatively prime to $p$, by Euler's Theorem $a^{\phi(p)} = 1 \bmod p$. Since $\phi(n) = \phi(p)\phi(q)$, we find that, mod $p$, $a^{k\phi(n)+1} = a^{k\phi(n)} \cdot a = a^{k\phi(p)\phi(q)} \cdot a = 1^{k\phi(q)} \cdot a = a$. And mod $q$, $a = 0$, so $a^{k\phi(n)+1} = 0^{k\phi(n)+1} = 0 = a$.

So $a^{k\phi(n)+1} = a \bmod p$ and $a^{k\phi(n)+1} = a \bmod q$, so by the Chinese Remainder Theorem, $a^{k\phi(n)+1} = a \bmod n$.

## 7.9 HOMEWORK PROBLEMS

1. If $m$ and $n$ are any two positive integers, show that $m/\gcd(m,n)$ and $n/\gcd(m,n)$ are relatively prime. [Hint: use the result of Euclid's algorithm.]

2. If $a$ and $b$ are relatively prime, and $bc$ is a multiple of $a$, show that $c$ is a multiple of $a$. [Hint: use the result of Euclid's algorithm.]

3. In mod $n$ arithmetic, the quotient of two numbers $r$ and $m$ is a number $q$ such that $mq = r \bmod n$. Given $r$, $m$, and $n$, how can you find $q$? How many $q$s are there? Under what conditions is $q$ unique? [Hint: $mq = r \bmod n$ iff there is an integer $k$ such that $qm + kn = r$. Divide by $\gcd(m,n)$.]

4. In the final step of Euclid's algorithm for finding $\gcd(m,n)$, we get $u$ and $v$ such that $um + vn = 0$. Is $|um|$ (which = $|vn|$) the least common multiple of $m$ and $n$?

5. Prove the general case of the Chinese Remainder Theorem. [Hint: $z_1$ is relatively prime to $z_2 z_3 \cdots z_k$.]

6. Show that each row of the $\mathbf{Z}_n^*$ multiplication table is a rearrangement of the 1 row. [Hint: multiply the row by the inverse of its first element.]

7. For what type of number $n$ is $\phi(n)$ largest (relative to $n$)?

8. For what type of number $n$ is $\phi(n)$ smallest (relative to $n$)?

9. Is it possible for $\phi(n)$ to be bigger than $n$?

10. If $n = p_1^{\alpha_1} p_2^{\alpha_2} p_3^{\alpha_3} \cdots p_k^{\alpha_k}$ where $p_i$ is prime, what is $\phi(n)$?

11. In RSA, what is the probability that something to be encrypted will not be in $\mathbf{Z}_n^*$?

12. Euler's Theorem Variant states that for all $a$ in $\mathbf{Z}_n^*$, $a^{k\phi(n)+1} = a \bmod n$. As stated in §7.8.1 *A Generalization of Euler's Theorem*, if $n$ is the product of two distinct primes, $a^{k\phi(n)+1} = a \bmod n$ for all $a$ in $\mathbf{Z}_n$. For what other forms of $n$ will this be true? For what forms of $n$ does it fail?

13. Prove that if $n = 2^{\alpha_0} p_1^{\alpha_1} p_2^{\alpha_2} p_3^{\alpha_3} \cdots p_k^{\alpha_k}$ where $p_i$ are distinct odd primes and $\alpha_i > 0$ for $i > 0$, then, mod $n$, 1 will have $2^k$ square roots if $\alpha_0 \le 1$, $2^{k+1}$ square roots if $\alpha_0 = 2$, and $2^{k+2}$ square roots if $\alpha_0 \ge 3$. [Hints: use the Chinese Remainder Theorem to show that a number is a square root of 1 mod $n$ iff it is a square root of 1 mod each of the prime power factors; show that 1 and $-1$ are the only square roots of 1 mod a power of an odd prime; finally, find the square roots of 1 mod a power of 2.]

# 8 MATH WITH AES AND ELLIPTIC CURVES

## 8.1 INTRODUCTION

We've seen a plethora of techniques for performing secret key and public key cryptography, and the number theory behind some of the public key algorithms. These public key schemes are based on the difficulty of factoring large integers, or the difficulty of calculating discrete logarithms over $\mathbf{Z}_p$*. While these problems still seem intractable, some significant progress has been made, and that makes cryptographers nervous.

So cryptographers have started exploiting somewhat different mathematical structures to use as the basis for cryptographic schemes. In this chapter we'll explore the mathematics needed to properly understand Rijndael/AES (described algorithmically in §3.5 *Advanced Encryption Standard (AES)*) and to vaguely understand elliptic curve cryptography. As with the previous chapter, this chapter requires no background other than intellectual curiosity, a more than vague remembrance of high school algebra, and trust that it's all understandable with a fair bit of thought, lots of patience, and several nights' sleep. We cover a great deal of material in a very few pages, so don't expect to skim through it all. We suggest you do the homework problems at the point they are referenced in the text.

If you skipped the last chapter, and not because you already knew it all, you almost certainly want to skip this chapter, or go back and read the previous chapter first.

## 8.2 NOTATION

In order to avoid cumbersome long formulas with lots of ellipses ("…"s, not conic sections), we'll be using $\Sigma$ notation for sums and $\Pi$ notation for products:

$$\sum_{i=m}^{n} a_i = a_m + a_{m+1} + \cdots + a_{n-1} + a_n$$

$$\prod_{i=m}^{n} a_i = a_m a_{m+1} \cdots a_{n-1} a_n$$

As an example, we can define exponentiation (for non-negative integer exponents) in $\Pi$ notation as

$$a^n = \prod_{i=1}^{n} a$$

(A sum with no components is 0, a product with no components is 1.) Sometimes, instead of low and high limits, we'll specify the components of a sum or product with a condition, e.g., $a^n = \prod_{1 \le i \le n} a$. When not otherwise specified, the index variable ($i$ in the previous examples) takes on integer values satisfying the condition or limits. Within this chapter, if a low limit is not specified, it is assumed to be 0. Sometimes we'll leave out the index variable altogether when it's clear what it is.

We'll also use a bit of set notation. A set is a collection of elements. We write $s \in S$ to indicate that $s$ is an element of the set $S$. If we know all the elements of a set, we can write it by listing its elements separated by commas and enclosed in curly braces, e.g. $\{0,1,2\}$ is a notation for the set comprising the first three non-negative integers. Listing an element more than once is the same as listing it once. If we know some property of the elements of a set, we can write it in terms of that property: $\{x \mid property(x)\}$ is the set of elements $x$ for which $property(x)$ is true. More generally, $\{expression \mid condition\}$ is the set of elements which can be written as *expression* subject to *condition*, e.g. $\{x^2+y^2 \mid x \in \mathbf{Z} \text{ and } y \in \mathbf{Z}\}$ is the set of numbers that can be expressed as the sum of two squares of integers. The difference of two sets $A$ and $B$, written $A-B$, is the set of elements in $A$ that are not in $B$.

## 8.3 GROUPS

Sometimes it's easier to understand something by generalizing it. So it is with $\mathbf{Z}_n$ (the integers mod $n$), which is the basis for so much of public key cryptography. Let's start by listing some of the properties we've seen while investigating $\mathbf{Z}_n^*$ (the integers relatively prime to $n$, mod $n$).

(A) **Associativity**. $(ab)c = a(bc)$.

(I) Existence of **identity**. There exists an element $e$ such that, for each $a$, $ea = ae = a$. In the case of $\mathbf{Z}_n^*$, this identity element is 1.

(N) Existence of **inverse**. For each $a$, there is an $a^{-1}$ such that $a^{-1}a = aa^{-1} = e$.

(C) **Commutativity**. For each pair of elements $a$ and $b$, $ab = ba$.

We'll define any structure $\langle G, \cdot \rangle$, a set of elements ($G$) and an operation ($\cdot$) taking two elements of $G$ and producing a single element of $G$, that satisfies properties A, I, and N to be a **group**. If the structure also satisfies property C, we'll call it a **commutative group**, also known as an **Abelian group**. Most groups used in cryptography are commutative groups.

When there is ambiguity, we'll specify the group operator explicitly; normally we'll use the same terminology and notation for the group operator as we use for multiplication, including exponentiation for repeated multiplication, where $a^0 = e$, $a^{-1}$ is the inverse of $a$, and $a^{-n} = (a^{-1})^n$.

Some examples of groups:

- $\mathbf{Z}_n^*$ (with multiplication mod $n$).
- $\mathbf{Z}_n$ with addition (mod $n$).
- $\mathbf{Z}$ with addition. (This is the simplest infinite group.)
- Permutations (i.e. rearrangements) of three objects, with the composition operator. (This is the simplest non-Abelian group.) See Homework Problem 1.

A **subgroup** of a group $G$ is a subset of $G$ that is a group under $G$'s operator. It is always the case that the identity element is a subgroup of any group, as is the group itself. It is also the case that the powers $\{g^n \mid n \in \mathbf{Z}\}$ of any element $g$ of the group is a subgroup, called the **cyclic subgroup** generated by $g$. If $G$ is finite, then we only need to include non-negative powers of $g$. A group $G$ is **cyclic** if it is its own cyclic subgroup, i.e. there is a $g \in G$ such that $G = \{g^n \mid n \in \mathbf{Z}\}$; $g$ is called a **generator** of $G$. A cyclic group can have many generators.

Some examples of subgroups:

- $\{1,9\}$ is a cyclic subgroup of $\mathbf{Z}_{10}^*$; 9 is the only generator.
- The even numbers form a cyclic subgroup of $\mathbf{Z}$ with addition. 2 is a generator.
- Permutations of the first three of four objects is a subgroup of the four-object permutations.

The **order** of a group is the number of elements in the group. We write the order of $G$ as $|G|$.

If $G$ is a finite group, and $H$ is a subgroup of $G$, then $|H|$ divides $|G|$. The proof involves creating a multiplication table, where the top row comprises the elements of $H$, starting with the identity. Successive rows consist of elements of the form $gh$, where $g$ is any element of $G$ not in any previous row of the table, and $h$ is in $H$. No two elements in the same row can be equal, because if $gh_1 = gh_2$, $h_1 = g^{-1}gh_1 = g^{-1}gh_2 = h_2$. (Notice how we just used the group properties of $G$.) It is easy to see that if $g$ is not already in the table, neither are any of the $gh$: if $gh$ is in the table, it is $g_0h_0$ for some $g_0$ in the first column and some $h_0$ in $H$; then $gh = g_0h_0$ so $g = g_0h_0h^{-1}$, so $g$ is in the $g_0$ row.

(Notice how we just used the group properties of $H$.) So all we have to do is fill the table row by row, each time choosing for the first column a $g$ in $G$ not already in the table. When there is no remaining $g$, we have a complete table, with each element of $G$ appearing exactly once, and each row containing $|H|$ elements. The number of rows is called the **index** of $H$ in $G$. So $|G|$ is the product of $|H|$ and the index of $H$ in $G$.

The **order** of an element is the order of the cyclic subgroup it generates. If $g$ has finite order, it is clear that the order of $g$ is the smallest positive integer $\lambda$ for which $g^\lambda = e$. Note that if $g^k = e$, then $\lambda$ divides $k$: if $k = q\lambda + r$ with $0 \le r < \lambda$, $g^r = g^{-q\lambda}g^k = e$, so $r = 0$.

***For the rest of this chapter, all the groups will be commutative.***

Pay attention to how we use this fact in the next paragraph (and see Homework Problem 2).

If we have two elements with finite order, we can find an element whose order is the least common multiple of those orders as follows. First, suppose $a$ and $b$ have orders $\lambda$ and $\mu$, respectively, and $\gcd(\lambda,\mu) = 1$. Since $\gcd(\lambda,\mu) = 1$, there are integers $u$ and $v$ such that $u\lambda + v\mu = 1$. Consider the cyclic group generated by $ab$. $(ab)^{u\lambda} = a^{u\lambda}b^{u\lambda} = eb^{u\lambda} = eb^{u\lambda}e = eb^{u\lambda}b^{v\mu} = b^{u\lambda+v\mu} = b$, so $b$ is in the cyclic group. Similarly, $(ab)^{v\mu} = a$, so $a$ is in the cyclic group. Since the order of each element of a group divides the order of the group, the order of the cyclic group (which is the order of $ab$) must be a multiple of $\lambda\mu$. And $(ab)^{\lambda\mu} = a^{\lambda\mu}b^{\lambda\mu} = ee = e$, so $ab$ has order $\lambda\mu$. Now if $a$ and $b$ have orders $\lambda$ and $\mu$, respectively, and $\gcd(\lambda,\mu) = \gamma$, let $\delta = \gcd(\gamma,\mu/\gamma)$ and $\varepsilon = \gamma/\delta$. Consider $a^\delta$ and $b^\varepsilon$. These have orders $\lambda/\delta$ and $\mu/\varepsilon = \mu\delta/\gamma$, respectively, and $\gcd(\lambda/\delta,\mu\delta/\gamma) = 1$, so the order of $a^\delta b^\varepsilon$ is $\lambda\mu/\gamma = \mathrm{lcm}(\lambda,\mu)$.

A corollary of this result is that for any finite Abelian group, there is an element whose order is the least common multiple of the orders of all the elements. We can produce such an element as follows: set our candidate to the identity, then repeatedly choose an element whose order is not a divisor of the order of the current candidate and replace the candidate with an element whose order is the lcm of the orders of the candidate and the newly chosen element; eventually we'll run out of elements and the candidate will have the desired order. We'll make use of this corollary in §8.4.2 *Finite Fields*.

## 8.4 FIELDS

We'll now consider two operators (+ and ·) at the same time, as we do with normal arithmetic on real numbers. The property we'd most like, because we're so used to it, and because it is so useful, is

(D) **Distributivity** of · over +. For all $a,b,c$, $a\cdot(b+c) = a\cdot b + a\cdot c$.