1. Introduction

Public-key cryptology was made possible by exploiting properties of multiplication modulo $n$. This allowed the construction of things generally believed to be one-way functions: computation of the function was fairly simple, but there was no quick way to compute an inverse of the function.

One example is $a^x \pmod{p}$ where $p$ is a fixed large prime and $a$ is a primitive root modulo $p$, i.e., a number whose powers take all $p - 1$ nonzero values modulo $p$. The binary expansion of $x$ gives a process for finding $a^x \pmod{p}$ that never needs to consider numbers larger than $p^2$ and requires a number of multiplications no more than a constant times $\log p$. Finding $x \pmod{p - 1}$ from $a^x$ modulo $p$ is known as the discrete logarithm problem. No method is known with a running time less than any fixed power of $\log p$.

The method used by RSA uses a number $n = pq$ that is a product of two large primes. The number $n$ is made public, but its factors $p$ and $q$ are kept secret. This means that finding the number of elements relatively prime to $n$, which is $(p - 1)(q - 1)$ is equivalent to factoring $n$, and this seems to be a difficult problem (in spite of the fluency you acquired at a young age in factoring numbers less than 100, factoring a number of 100 digits will keep a computer busy for some time). Knowing $p$ and $q$, it is easy to construct a pair of numbers $e$ and $f$ with $ef \equiv 1 \pmod{(p - 1)(q - 1)}$, so that $y = x^e$ and $x = y^f$ are inverse operations modulo $n = pq$, but it does not seem possible to find $f$ from $e$ and $n$ without first factoring $n$. This allows $e$ to be used as a public key with $f$ saved as a private key that can be used only by its owner to decrypt messages constructed with the public key.

The significance of multiplication modulo $n$ in these examples is that it uses the process of ordinary multiplication that we all know how to do and that computers can do quickly and reliably. The additional work to find $x^e$ or $a^x$ involves only a small scheme of repeated multiplications that is simple programmed. Hand computation can be used to illustrate how the method works, but it would be tedious to do the work required for a useful example. However, computers don’t seem to mind.

Any operation that could be reduced to ordinary arithmetic would do just as well. The S-boxes of common symmetric secret key systems involve simple computation that are typically described in terms of Boolean operations, but they are chosen because of their ease of computation combined with the ability to invert them by using the same operation (depending on a secret key known to both “Alice” and “Bob”). One complication is that these operations are essentially general permutations, so the inverse of a product requires multiplying the inverses of the factors in the reverse
order. This defeats most of the protocols designed for key-sharing or digital signatures.

It would be desirable to have a commutative system (like one based on ordinary multiplication) in which doing $A$ and then doing $B$ had the same effect as doing $B$ and then $A$.

2. Elliptic curves  
Fortunately, a huge family of examples of commutative algebraic systems have been known by mathematicians for some time. My source is Lawrence C. Washington: “Elliptic Curves: Number Theory and Cryptography”, Chapman & Hall/CRC 2003 (ISBN 1-58488-365-0). Although fairly light on the details of the cryptographic applications, he quotes some results claiming that a 313 bit elliptic curve system afforded comparable security to 4096 bit RSA, and that a Palm Pilot was able to generating a 512 bit RSA key took 3.4 minutes, while a comparable 163 bit key for an elliptic curve system took 0.597 seconds. The gain for using more advanced mathematics is a more compact system that allows faster computation.

An elliptic curve may be taken to be the set

$$\{ (x, y) : y^2 = x^3 + Ax + B \}$$

where $x$ and $y$ live in some number system (i.e., a field). Here, $A$ and $B$ are fixed constants in the same system where we are looking for the solutions. For our purposes, we will work in the integers modulo a prime, but some important questions of mathematical interest ask about the rational solutions of such equations when $A$ and $B$ are given integers. In particular the distinction between equations with only finitely many rational solutions and equations with infinitely many rational solutions is important.

In order to be able to define an operation of addition of points on the curve, it is necessary to add a “point at infinity”. The geometric constructions underlying the operation are based on intersecting the curve with lines, and we want to avoid difficulties related to the fact that parallel lines don’t intersect. The system of projective geometry was invented in the nineteenth century to control such exceptions. It is still the main approach to the study of curves defined by algebraic equations, but higher dimensional objects and number-theoretic properties of curves are now studied from a more abstract viewpoint. We break our study of elliptic curves to introduce the tools of projective geometry.

3. Projective Geometry  
To study the plane with coordinates $(x, y)$ we embed it in three dimensions as the plane where $z = 1$. All properties of the plane can still be described using $x$ and $y$, and geometric constructions look the same. However, the points $(x, y, 1)$ of the plane determine a line through the origin consisting of points with coordinates $(tx, ty, t)$ for all values of $t$, and this line meets our plane in only one point. Similarly, a line in our plane with equation $ax + by = 1$ can be extended to a plane through the origin in three dimensional space with equation $ax + by = z$. The idea of projective geometry is to use these lines and planes in three dimensional space in place of the points and lines of ordinary plane geometry. The benefit of this is that any two different planes through the origin intersect in a line through the origin. That is, in projective geometry, any two different lines intersect. This is possible because there are more
lines through the origin than we constructed from our original plane. Any line in the plane \( z = 0 \) has no point on the plane \( z = 1 \), and points with small values of \( z \) give lines meeting \( z = 1 \) in points whose \( x \) and \( y \) coordinates are large. This has led \( z = 0 \) being called a **line at infinity** for the original plane. It contains lots of points: one for each direction of parallel lines in the original plane. Parallel lines now meet at a particular **point at infinity** determined by the direction of the line.

To work in projective geometry, every expression in original coordinates \( x, y \) should be rewritten with \( x/z \) in place of \( x \) and \( y/z \) in place of \( y \). We can then multiply by a power of \( z \) to get a homogeneous polynomial expression in \( x, y, z \). For example, \( y^2 = x^3 + Ax + B \) becomes \( y^2/z^2 = x^3/z^3 + Ax/z + B \), and then \( y^2z = x^3 + Axz^2 + Bz^3 \). Every term in the final expression has degree 3. Now, if \( z = 0 \), the equation demands that \( x = 0 \), so the only point at infinity on the curve is the point on lines parallel to the \( y \)-axis.

The problem of determining the intersection of two plane curves is simplified by using projective geometry. The study of homogeneous polynomials in three variables has a theory of multiplicity. A polynomial of degree \( m \) and one of degree \( n \) with no common polynomial factor meet in exactly \( mn \) points when counted with multiplicity. The simplest example of multiplicity is the intersection of a curve of degree 2 (you may think of a circle, if you like) and its tangent line. The unique point of intersection is counted twice in this case. Using the equation of the line to eliminate one of the variables leads to a homogeneous equation of degree 2 in the remaining two variables, and this quadratic equation has two roots that become a single root of multiplicity 2 when the line is tangent. It is also possible that the quadratic equation will have no solutions in the desired number system, but the system can be extended to introduces new numbers to describe all points of intersection.

4. **An example of an elliptic curve**  
Consider \( y^2 = x^3 + x + 1 \mod 7 \). The squares modulo 7 are 1, 2, and 4, and if we write \( f(x) = x^3 + x + 1 \), then, modulo 7, \( f(0) = 1, f(1) = 3, f(2) = 4, f(3) = 3, f(4) = 6, f(5) = 5, f(6) = 6 \). Comparing these two lists, we see that there are 4 points on the curve modulo 7: (0, 1), (0, 6), (2, 2), (2, 5). The **point at infinity** gives a fifth point. The vertical lines \( x = 0 \) and \( x = 2 \) meet the curve at two ordinary points and the point at infinity. The line at infinity has the point at infinity as a **triple point**. The line \( y = 4x + 1 \) meets the curve where
\[
x^3 + x + 1 = (4x + 1)^2 = 2x^2 + x + 1 \mod 7
\]
so the point (0, 1) is a double point and (2, 2) has multiplicity 1. Constructing the tangent at (0, 1) by methods of Calculus leads to \( y = x/2 + 1 \) which is equivalent to our equation modulo 7. The line \( y = 2x + 1 \) meets the curve once at (0, 1) and twice at (2, 5).

The addition law on an elliptic curve is defined so that the point at infinity plays the role of zero, and the sum of the three points of intersection with any line is zero. Some careful algebra is needed to show that this **always** has the expected properties. For the simple example that we have described, we have checked that the successive multiples of (0, 1) are (0, 1), (2, 5), (2, 2), (0, 6).
5. **Singularity**  This doesn’t work for every \( y^2 = f(x) \) with \( f(x) \) a cubic polynomial. If \( f(x) \) has a **multiple root** at \( x = a \), the point \((a, 0)\) behaves in a special way, and the mathematical word for such special behavior is “singular”. Every line through \((a, 0)\) must have this point as (at least) a double point, so it can meet the curve at no more than one other point. Unless something is done to treat \((a, 0)\) as a **curve** with one point for each direction, it will break the rule defining addition on the curve. There is an algebraic way to do this, and it is important, but it doesn’t lead to the same sort of operation as one finds in elliptic curves. Since such curves are easily found, they can be easily avoided when seeking a vehicle for cryptographic applications.

6. **Counting points**  Aside from the 1, 2 or 4 points that are either the point at infinity or have \( y = 0 \), other points come in pairs with the same value of \( x \). It seems reasonable that values for \( f(x) \) would be squares roughly half the time, so the curve will wind up with about \( p + 1 \) points, including the point at infinity, modulo \( p \). A precise form of this has been proved: the difference between the number of points and this estimate is never more than \( 2 \sqrt{p} \). Although the computation of the addition law is more complicated than multiplication modulo \( p \), it is still easily programmed. In all other ways, all that was used about multiplication was its basic algebraic properties, and these are shared with elliptic curves. Since we are no longer locked into a size of \( p - 1 \) after choosing \( p \), much greater flexibility is available. Since there are many curves modulo \( p \), it will be possible to find curves whose number of points has other desirable properties. This is what allows lies behind the ease of creating examples.