NuCypher Network: Pricing Protocol & Economics[*]

(Dated: September 6, 2020)

[*] arjun@nucypher.com

I. OVERVIEW

A. Motivation

The first products offered via the NuCypher network, dynamic access control and secrets management, are leveraged by developers to constitute a component of an application or information system's underlying technology infrastructure. Through novel cryptography and a distributed network of independent service-providers, applications that have integrated NuCypher's runtimes and interfaces may then support end-to-end encrypted data sharing between end-users and/or endpoints. These encrypted data sharing workflows are scalable, redundant, extensible, censorship-resistant, and verifiably protect the privacy of the application's end-users, a combination of value propositions that is unique in the digital service marketplace at the time of writing.

Pricing the NuCypher service appropriately and expediently is not a straightforward exercise. Although end-to-end encryption has become so prevalent on mainstream digital platforms that in 2020 at least 4 billion user accounts are protected in this manner [1][2][3][4], it is not typically outsourced to a third-party in a commercial sense, and hence no well-known market price point or price structure for an 'end-to-end encryption service' exists. The products most similar to NuCypher's access control and secrets management, for which publicly available pricing exists, are cloud-based Key Management Systems (KMS), such as AWS KMS or Azure KeyVault. These provide a useful reference point for both the viability of certain price points and customer expectations/anchoring to certain price structures. Ultimately, cloud KMS products confer a quite different set of benefits to customers – they do not guarantee the protection of end-user privacy, nor resistance from censorship – and therefore competition-based pricing development is not a comprehensive strategy.

Difficulties notwithstanding, appropriate pricing is critical for the network to flourish; to grow and retain the network's user base, to support and sustain the network's service-providers, and to maintain and improve the aforementioned value propositions. On the demand side, the service must be affordable to as many addressable customer segments as possible (or better, that price quotes match or improve upon their perceived value of the product). On the supplier side, the revenue generated for service-providers must sustain their operations at an achievable stage of network adoption (that is, a stage reached within a survivable timeframe). Thus, much of pricing development is evaluating these two primary requirements, and squaring of the natural trade-off between them.

B. Network externalities & two-sided market model

NuCypher's pricing development must also account for the presence and evolution of network externalities. Customers benefit from increases to independent participation on the service-provider side, through, at the very least, gains in network security, redundancy and the geographical diversity of nodes (which can improve round-trip latency and choice of regulatory jurisdiction). Service-providers benefit from growth in the customer base, usage and the commensurate volume of fees paid into the network. Put simply, participation on one side of the network affects participation on the other – and hence we may model the network as a *two-sided market*[1]. This model enables

[1] A two-sided market is generally defined as one where the volume of transactions through the intermediary platform (or network) depend on the pricing *decomposition* – i.e. the extent to which pricing and other economic levers are skewed to favor one side.

the NuCypher community to draw insights from literature, and generate theoretical predictions of impact of certain pricing (and subsidy) structures. For example, Rochet & Tirole's modification of the standard Lerner-index[2] may be leveraged to optimally structure the network's pricing:

$$\frac{p^i - (c - p^j)}{p^i} = \frac{1}{\eta^i} \tag{1}$$

Unlike the commercial entities behind centralized two-sided markets (e.g. Uber, Microsoft/Xbox, Walmart.com), the NuCypher protocol does not extract a commission from service-providers or customers for transacting and exchanging value on the network. This means we cannot use off-the-shelf formulae that describe profit maximization of the intermediary platform itself. Nonetheless, if any form of universal pricing is enforced by the NuCypher protocol (see next two sections), it is inevitable that the constraints will skew pricing to effectively favour one side of the market. Moreover, just like the aforementioned Uber, Microsoft and Walmart corporations, the NuCypher protocol heavily subsidizes service-provider operations. Subsidies, though 'paid for' by a third group in NuCypher's case (through the dilution of passive holders of the native token), are still an artificial economic influence and also skew the network's effective pricing (though not necessarily in the same direction).

In other words, the NuCypher protocol includes mechanisms that efficaciously redistribute value from one side of the market to the other. Endogenous mechanisms of this sort, whether they arise from corporate strategy (in the case of a profit-maximizing digital platform) or governance-driven protocol updates (in the case of a minimally extractive decentralized network), are designed and implemented with the explicit intent of maximizing the volume of market transactions. Not only does this objective function serve as a decent approximation of profit maximization, there are similar levers available to adjust the pricing decomposition – for example, lowering the maximum fee rate chargeable by all service-providers, thereby skewing the pricing towards the customer side of the market.

Hence, in evaluating characteristics of the NuCypher network as part of pricing development (e.g. the price elasticity of demand), one must account for these skews, consider the impact of positive (and negative) network externalities, and treat the NuCypher network as a variety of two-sided digital market.

C. Independent Pricing (Free Market)

Given the permissionless, self-determining nature of service provision in decentralized networks, and the lack of profit maximization and marginal costs as intermediary attributes of the network itself, protocol designers may be tempted to leave the difficult task of pricing up to each individual service-provider. The consequences of mispricing will then fall on the shoulders of each independent operator – they'll either adapt or die, but the network as a whole, boasting a diversity of prices and a replenishable array of service-providers, will survive. This is a reasonable assertion. Yet, a free market approach comes with a litany of issues:

[2] Where η^i is the elasticity of volume (demand) with respect to price on market-side i, p^i is the per-transaction price on market-side i, p^j is the per-transaction price on market-side j, and c is platform or intermediary's marginal cost of facilitating the transaction. The intuition for this formula is that a given increase in the per-transaction price on side i may result in the loss of a transaction. This loss is fully captured in the *opportunity cost* $(c - p^j)$, which includes an additional loss of revenue that may otherwise be extracted from side j [5].

- **Price instability**. The adoption of NuCypher by application developers necessitates irretrievable integration costs and a time lag between making the decision to integrate and commencing network usage. Adopters need a guarantee that the price quotes, on the basis of which they decide to leverage the service, do not change to such an extent by the time integration is complete that their application's usage of the service is rendered unaffordable or undesirable. A free market, where service-providers can update their pricing whenever and however they want, does not provide this guarantee.

- **Multi-provider service coordination**. Typical usage of the NuCypher service necessitates concurrent work by multiple service-providers in order to be maximally secure and redundant. A free market, and higher variance in price points, increases procurement friction for customers and probabilistically increases the uniform fee they must pay to hire multiple service-providers. This feature of the NuCypher service also exacerbates the price stability problem, as it increases sensitivity to changes in pricing effectuated by some fraction of the service-provider population.

- **Undercutting & centralization**. Fully independent pricing increases the scope of undercutting strategies, which generally favor deeper-pocketed or larger service-providers (through economies of scale and other advantaged) and therefore may cause or accelerate centralization trends. This problem is preempted by Ethereum; although gas pricing allows customers to expedite the execution of a transaction with a higher bid, the cost of computation and storage is fully standardised across opcodes (operations) to avoid further amplifying the advantage of more efficient, higher-resourced nodes [6].

- **Lack of scope for product differentiation**. There is very little horizontal differentiation between the product offered by service-provider running NuCypher software and the product offered by another[3]. Interoperability requirements between service-providers mean there is limited scope and motivation to develop a differentiating feature that might be offered exclusively to users. This means that permitting price diversity is unlikely to benefit the network by fostering greater competition, raising service quality, or diversifying features.

- **Lack of scope for price discrimination**. Although in theory a service-provider could engage in price discrimination based on the funds in a customer's wallet, this strategy is unlikely to succeed given the near-zero information asymmetry and near-zero switching costs. There is also research asserting that price discrimination in two-sided markets weakens rather than bolsters competition [7]. Permitting price discrimination is also unlikely to benefit the service.

D. Quasi-universal Pricing

Universal pricing – a single price point for a standardized service unit, dictated by the protocol and adhered to by both users and service-providers without exception – addresses the problems listed above.

A major disadvantage of this approach is that the first few attempts to parametrize the price points will almost certainly be non-optimal. Prior to network launch there are major unknowns; including the value perception of the service and the prospective customers whose infrastructure budgets constrain their decision-making. Post-launch,

[3] There may be some vertical differentiation through the geographical location of nodes and the associated latency and regulatory benefits. Whether this is a feature desired by network user remains to be substantiated.

this insight will slowly surface over time, anecdotally (e.g. customer integration announcements) and via network statistics (e.g. growth in daily transactions over time). Even so, unknowns about the *next set* of would-be customer attributes will persist, which means a more robust solution is preferable:

Firstly, by instituting a global fee range ('quasi-universal pricing'), within which all service-providers' individually chosen minimum price points must fall. This broadens the protocol's corrective power. Service-providers holding diverse views on price optimality will reflect this in their individual parameters, and may toggle these price points in response to changing circumstances or new market information.

Secondly, by actively encouraging and enabling the regular modification of the global fee range through governance channels. Collectively setting prices gives participants a window into the operations of others, as well as their views on other critical considerations for pricing – e.g. the maximum price a target customer segment will tolerate. This information surfaces via provider-authored proposals, community push-back and debate.

These two mitigating modifications to universal pricing harness some of the crowd wisdom of the free market, without succumbing to the disadvantages of fully independent pricing explored above. Tuning the width of the global fee range to account for this trade-off is

E. Service Unit & Payment

Regardless of the pricing structure, NuCypher's base unit of service should be defined, standardized and universally followed. At network launch, a customer will pay for dynamic access control and secrets management with the following attributes, together comprising the minimum chargeable unit:

- Per sharing policy
  - Unique data controller (*Alice*)
  - Unique designated recipient (*Bob*)
  - Unique file directory address[4] (*label*)
- Per 24-hour period
- Per service-provider
  - Unique worker address (*Ursula*)

Note that this service unit does not include the number of access or re-encryption requests made to the Ursulas managing the sharing policy whilst it is active, nor does it take into account the computational overhead of running a service-provider. Although this service unit delineation is imperfect, the primary price-determining input for this service unit is rather simple: *duration*. This input aligns with the primary cost-determining input for service-provider operations, in that the main overheads are infrastructural (e.g. hosting) and periodic chain interactions (i.e. gas costs), both of which rise linearly with time.

[4] No limit on the number or size of files stored at this address.

F. Reconciling demand-side & service-side constraints

Given our definition of the network's primary service unit in the previous section, we may derive expressions, constructed from an overlapping variable set, to describe and process key budgetary constraints for the demand-side (users/developers) and service-side (service-providers).

On the demand-side, we describe the annual cost per user ($ACPU$). This is a critical balance sheet overhead for digital applications and information systems, which will likely comprise the bulk of NuCypher adopters.

$$ACPU = F_{pp} \cdot P_a \cdot P_d \cdot n \tag{2}$$

F_{pp} is the standardized cost of a single service unit – the fee Alices pay per sharing policy, per period, per Ursula[5].
P_a is the average number of sharing policies generated by a single end-user of the application, annually.
P_d is the average duration of all sharing policies created by all end-users of the application, in periods.|
n is the average number of service-providers assigned to each sharing policy.

If we include the primary third-party costs of using the service, then the $ACPU$ is modified to:

$$ACPU = P_a \cdot \big((F_{pp} \cdot P_d \cdot n) + G_p\big) \tag{3}$$

G_p is the cost in Ethereum gas of creating a single sharing policy ('granting'). This cost may be internalized into the transaction through a discounted price[6].

For the service-side, we describe the annual revenue (AR) generated by participating in the NuCypher network. This a standard balance sheet metric for any commercial entity, and therefore all NuCypher service-providers.

$$AR = \frac{(ACPU - P_a \cdot G_p) \cdot U_t}{S_t \cdot n} = \frac{F_{pp} \cdot P_a \cdot P_d \cdot U_t}{S_t} \tag{4}$$

U_t is the total number of end-users, spread across all adopting applications and systems, using the NuCypher network at a given moment in time. S_t is the total number of service-providers supporting the NuCypher network[7].

Using the variable $ACPU_{max}$ – the highest sum, per end-user per year, an addressable customer segment is willing and able to pay, we may describe the maximum fee per service unit from a demand-side perspective F_{Dpp}.

[5] For this exercise we assume this parameter is a global invariant enforced by the network protocol.

[6] It must be noted that since gas costs are per policy, there are scenarios of end-user behavior in which the cost of gas exceeds the total budget for those end-users' sharing policies. In other words, fees may absorb some of the cost of granting, but only until a certain point (with respect to Pa). This discount also depends on the temporally variable cost of gas, and the acceptable maximum duration for the grant function to execute (i.e. the choice of gas price).

[7] It is assumed for the purposes of this exercise that all service-providers receive an equal cut of total work allocation and fee revenue. In reality work allocation is probabilistic and weighted by relative stake size.

$$F_{Dpp} = \frac{ACPU_{max}}{P_a \cdot P_d \cdot n} \tag{5}$$

If fees are discounted to absorb the gas costs of creating a sharing policy, then the highest fee is modified to:

$$F_{Dpp} = \frac{ACPU_{max} - (P_a \cdot G_p)}{P_a \cdot P_d \cdot n} \tag{6}$$

Using the variable AR_min, the lowest sum of yearly revenue a service-provider is willing or able to tolerate, we can describe the minim fee per service unit from a service side perspective F_{Spp}[8].

$$F_{Spp} = \frac{AR_{min \cdot S_t}}{P_a \cdot P_d \cdot U_t} \tag{7}$$

To illustrate the utility of these expressions, we will plug the following fiat estimates into equations 5 and 7. These are dummy figures and will change with use case analysis, increasing insight into NuCypher customer priorities/behavior/budgets, and post-launch network statistics.

$ACPU_{max}$: $1 developer budget per end-user per year for the NuCypher service
AR_{min}: $1,000 earned per year by each NuCypher service-provider

S_t: 50 total service-providers
U_t: 100,000 total users
P_a: 52 policies per year (mean)
P_d: 90 periods per policy (mean)
n: 5 service-providers per policy (mean)

These estimates yield a demand-driven F_{Dpp} of $4.27E-05 (XXX GWEI) and a service-side F_{Spp} of $1.07E-04 (XXX GWEI). If we use F_{Dpp} as our universal price point, ceteris paribus, service-providers will earn $400 annually. If we use F_{Spp} the average cost per user will be $2.50.

An important variable we can now project is the number of total network users – i.e. in order to see what level of adoption the network needs to reach for the dummy constraints $ACPU_{max}$ and AR_{min} to *both* be satisfied.

$$\frac{ACPU_{max}}{P_a \cdot P_d \cdot n} = \frac{AR_{min \cdot S_t}}{P_a \cdot P_d \cdot U_t} \tag{8}$$

[8] Note that although n cancels out in this expression, it still affects F_{Dpp} above, which it is a determinant of the demand function (the uptake of the service at various prices), and therefore indirectly affects service-side revenue.

$$U_t = \frac{AR_{min} \cdot S_t \cdot n}{ACPU_{max}} = 250,000 \tag{9}$$

[1] Number of monthly active whatsapp users worldwide from april 2013 to march 2020, 2020. URL https://www.statista.com/statistics/260819/number-of-monthly-active-whatsapp-users/#:~:text=Number%20of%20monthly%20active%20WhatsApp%20users%20as%20of%202013%2D2020&text=As%20of%20March%202020%2C%20WhatsApp,billion%20MAU%20in%20February%202016.

[2] Number of snapchat users worldwide from 2018 to 2023, 2020. URL https://www.statista.com/statistics/626835/number-of-monthly-active-snapchat-users/.

[3] Number of unique viber user ids from june 2011 to march 2020, 2020. URL https://www.statista.com/statistics/316414/viber-messenger-registered-users/.

[4] Number of estimated skype users registered worldwide from 2009 to 2024, 2020. URL https://www.statista.com/statistics/820384/estimated-number-skype-users-worldwide/.

[5] Jean-Charles Rochet and Jean Tirole. Two-sided markets: a progress report, 2006. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1756-2171.2006.tb00036.x.

[6] Renlord Yang, Toby Murray, , Paul Rimba, and Udaya Parampalli. Empirically analyzing ethereum's gas mechanism, 2019. URL https://arxiv.org/pdf/1905.00553.pdf.

[7] Qihong Liu and Konstantinos Serfes. Price discrimination in two-sided markets, 2007. URL https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1022422.