NetCheque and NetCash are service marks of the University of Southern California. All other product or service names are trademarks or service marks of their respective owners.

## *NetCheque℠, NetCash℠ and requirements for network payment*

B. Clifford Neuman
University of Southern California
Information Sciences Institute

## *Requirements for payment services*

Secure, reliable, flexible, scalable, efficient, and unobtrusive payment methods are required as a basic service of the Internet and must be integrated with existing and evolving applications.

## *Requirements: Reliability*

- Commerce will depend on the availability of the billing infrastructure.
- The infrastructure may be a target of attack for vandals.
- The infrastructure must be highly available and should not present a single point of failure.

## *Payment for Access to Internet Services*

- Many services envisioned for the Internet will only be realized if service providers are compensated for the services they provide:
  - Access to copyrighted materials
  - Database searches
  - Consumption of system resources
  - Merchandise delivered outside the system
  - Services provided outside the system

## *Requirements: Security*

- Information services are provided today on relatively open networks.
- Payments involve actual money; such systems will be a target for criminals.
- The infrastructure supporting electronic commerce must be usable on open networks and resistant to attack.

## *Requirements: Flexibility*

- Different models for different situations:
  - credit card, cash, check
- Different assurances are provided:
  - accountability, anonymity, risk
- There is a need for a common framework.
- Convertibility is needed across models

Copyright © 1995 University of Southern California

*Requirements: Scalability*

- The payment infrastructure should support multiple independent accounting servers and should avoid central bottlenecks.
- Users of different accounting servers must be able to transact business with one another and the funds must be automatically cleared between servers.

*Requirements: Unobtrusive*

- Users should not be constantly interrupted to provide payment information.
- However, users do want to control when, to whom, and how much is paid.
- Users must be able to monitor their spending.

*Three forms of payment*

- Secure presentation
- Electronic currency
- Credit-debit instruments

*Requirements: Efficiency*

- Frequent payments for small amounts must be supported (micropayments).
- Performance must be acceptable, even when multiple payments are required.
- Merchants and payment servers must be able to handle the load.
- Per-transaction cost must be small enough that it is insignificant.

*Requirements: Integration*

- Payment systems must be tied to the existing financial infrastructure.
- Applications must be modified to use the the payment infrastructure.
- Payments should be supported by common protocols that underlie applications.
- A common framework should support integration of multiple payment methods.

*Three forms of payment*

Secure presentation

- Uses traditional credit card numbers
- As safe as the phone
- Often no customer signature
- Potentially huge customer base
- Little need for infrastructure

Copyright © 1995 University of Southern California

*Examples of secure presentation*

- NetScape's commerce server
- Secure Mosaic
- CyberCash's initial offering
- iKP (1 and 2)

*Three forms of payment*

Electronic currency

- Potential for anonymity
- Potential for off-line operation (debated)
- Requires extensive matching capabilities

*Considerations: electronic currency*

- Integrity and security
- On-line vs. off-line
- Is tamper resistant hardware needed
- Who is at fault for counterfeit currency
- Kinds of anonymity

(continued)

*Considerations: secure presentation*

- Integrity and confidentiality
- Who sees the account number
- Is the payment order signed
- Is the merchant legitimate
- Real time authorization
- Transaction cost

*Examples of electronic currency*

- Chaum's DigiCash
- Mondex
- USC/ISI's NetCash[SM]

*Considerations: electronic currency*

(continued)

- Transaction cost
- Storage requirements
- Getting money in and out of the system
- Backing of the currency

Copyright © 1995 University of Southern California

## *Three forms of payment*

Credit-debit instruments

- Only authorized individuals can spend from particular accounts
- Records of transactions maintained
- Requires new infrastructure

## *Considerations: credit-debit*

- Security for the instrument
- Compromise of user's keys
- Who incurs risk
  - From compromised account
  - From unpaid bills
- Real time authorization
- Transaction cost

## *Sample Payment Systems (not a complete list)*

- Fist Virtual
- NetScape
- Secure Mosaic
- CyberCash
- iKP
- Open Market (integration)
- NetCheque
- NetBill
- FSTC's Electronic Check
- DigiCash
- NetCash
- Mondex

## *Examples of credit-debit model*

- First Virtual
- USC/ISI's NetCheque[SM]
- CMU's NetBill
- FSTC's Electronic Check project

## *Other forms of payment*

Direct Transfer

- Customer initiates transfer of funds to account of the merchant

Collection agent

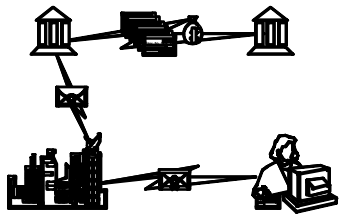
- Payment collected by third party who informs merchant that payment has been made.

## *USC/ISI's NetCheque[SM]*

- No prior arrangement between customer and merchant.
- A check authorizes the payee to transfer funds from the payor's account.
- Multiple currencies per account.
- Payments clear through multiple payment servers.

Copyright © 1995 University of Southern California

## *Flow of payment instrument*

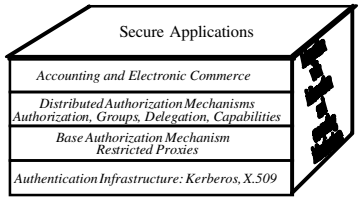


## *NetCheque: MIME Encoded*

```
--NetCheque(SM)V1.0
Content-Type: APPLICATION/X-NETCHEQUE
Content-Transfer-Encoding: BASE64
Content-Description: Pay 10.00 NCU to marketplace@NETCHEQUE.ISI.EDU

AAAAAQAAAA5OZXRDaGVxdWVfVjEuMAAAAA1TT0ZUV0FSRV9WMS4xAAAAAQED
NTE4AzI2N2GCAQcwggEDoAMCAQWhExsRTkVUQ0hFUVVFLklTSS5FRFWiKTAn
oAMCAQGhIDAeGwlOZXRDaGVxdWUbEW5ldGNoZXF1ZS5pc2kuZWRlo4G7MIG4
oAMCAQGhAwIBAaKBqwSBqEILdnGDj8taheicu2b3DK+0qYB+ayEtyZUdVsyC
RVFVRS5JU0kuRURVAAAABQAAAAIBM05DVQExATEAAAAEAjU5AAAACwk4MDAw
MzQ4NzkJODAyMTk0Nzk4AAAACQIxNUNsaWZmb3JkX051dW1hbgAAAAEBMQEx
AAAAHW1hcmtldHBsYWNlQE5FVENIRVFVRS5JU0kuRURVAAAAAA==
--NetCheque(SM)V1.0--
```

## *NetCheque security*



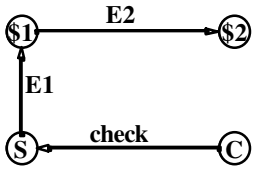

## *NetCheque representation*

- Internal representation is opaque
- Important fields:
  - Account and accounting server
  - Amount, payee, expires
  - Memo: Customer info, merchant info
  - Signatures and endorsements
- MIME encoded for use by applications
  - Base 64 encoding for opaque part

## *NetCheque security*

- Check has plaintext part and signature
- Endorsements are separately signed and linked to a particular check
- Signature component is modular
  - Current implementation is Kerberos proxy
    Signature verifiable by customer's bank
  - Can accommodate RSA or DSS signatures

## *Clearing the check itself*



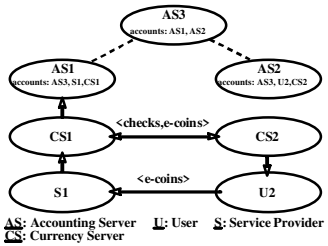

check: [ckno,amount,S]C
E1: [ckno,amount,S]C [dep ckno to $1]S
E2: [ckno,amount,S]C[dep ckno to $1]S[dep ckno to$2]$1

Copyright © 1995 University of Southern California

## *NetCheque status*

- Initial NetCheque release December 1994.
- Integration with WWW browsers, and merchant support released May 1995.
- NetCheque trial underway with play money (NCU=NetCheque Currency Unit)
- NetCash to be available mid 1995
- Contact Information: NetCheque@isi.edu
  - URL: http://nii.isi.edu/info/NetCheque

## *Anonymous payment using NetCash℠*




## *Payment Integration: PPV*

- PPV is a payment protocol developed for use by NetCheque and NetCash.
- PPV is layered over web protocols and requires no changes to web browsers.
- PPV is multi-mechanism, supporting other forms of payment instruments.
- PPV is a temporary measure until direct support for payment is provided.

## *USC/ISI's NetCash℠*

- Users purchase currency from currency server using NetCheque - deposits to currency server's account back the currency
- Supports weakly anonymous payment
  - Cash can be exchanged for new cash anonymously
  - Customer chooses the currency server
- Multiple currency servers, NetCheque used to clear cross-server payments

## *Payments and applications*

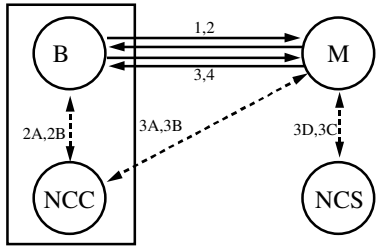
- Applications must support payment.
- Need thresholds for automatic payment.
- Confirmation for larger payments.
- Feedback to users for current spending.

## *PPV payment protocol*

- Browser requests document from merchant.
- Merchant returns preview and payment information to browser
- Browser connects to PPV client (a local HTTP server) and authorizes payment.
- Client sends payment or information needed to retrieve payment to merchant.
- Merchant returns document to browser.

Copyright © 1995 University of Southern California

## *PPV protocol*




## *Tighter integration needed*

- Functionality currently uses CGI.
- Fold scripts into HTTP server itself.
  - Port 1564 allocated for PPV
- Browsers need to subsume scripts.
- Confirmation should be distinguishable from remote web page.

## *Multiple financial service providers*

- If financial service providers compete on the basis of incompatible systems, payment services will be difficult to integrate and use.
- Instead, multiple service providers will exist and it will be possible to clear payments between providers.
- Designs for payment services should allow for multiple financial service providers.

## *PPV document format*

- Documents consist of
  - Main document, and
  - Document header, including
    - Payee
    - Price
    - Payment methods accepted
    - and more

## *Better integration with Mosaic*

- In-band negotiation for payment as a defined part of http or other specification:
  - As an SHTTP extension, or
  - As part of Secure Socket Layer, or
  - Additional tags in HTML

## *Sharing the risks and rewards*

- Agreements will be needed to allocate risk between financial service providers.
- Agreements on the splitting of fees will also be required.
- Service providers will compete on the basis of established trust, fees, and value added services.

Copyright © 1995 University of Southern California